# FM04a: Reimbursement

**September 23, 2005**

# Table Of Contents

## 1.1. Introduction

*FM04a: Reimbursement* topic is comprised of three batch processes and two pages. The objective of this topic is to reduce the amount of State funds used towards the cost of ongoing care. Two additional funding sources are available to the State for reimbursement of these costs. These sources are Trust Account money and Federal money. Calculation of how much the State may reimburse itself from these sources will be a monthly process.

The first step in the reimbursement process is to withdraw money from a child's trust account. This is accomplished by the Trust Account draw down batch, which will access those benefits that have been designated by the Trust Account manager as available for State reimbursement. Any money set aside for Personal Needs will automatically not be used towards ongoing care. Other benefits can likewise be indicated as unavailable for State reimbursement. This designation occurs through the Trust Account window (See *FM07: Trust Accounts*). All other benefits will be used towards ongoing care costs. Remaining costs not covered by the trust account is considered to be the non-trust amount. In addition, any overpayments to provider, for which money was taken from the trust account, must have money manually returned to the child's trust account. The Overpayment With Trust Accounts Report is available to assist the Trust Account Manager with this task.

The second step in the reimbursement process is to designate payments as either Federal Financial Participant (FFP) or non-FFP. This determination is based on the Eligibility status determined through the either the Eligibility page or the Adoption Eligibility page (See *FM03: Eligibility*). The Eligibility page will display one of four statuses: 1) Eligible and Reimbursable 2) Eligible and Non-Reimbursable, 3) Ineligible or 4) Pending. The Adoption Eligibility page will display one of three statuses: 1) Federal, 2) State or 3) Pending. Because a change in eligibility status may occur mid-month the number of days earmarked for each status must be determined based on the payment begin date and payment end date.

The third step in the reimbursement process is to calculate the percentage of the non-trust amount that may be federally reimbursed. For all providers, the percentage of federally reimbursable cost is based on the percentage stored on the Maintain Title IVE-FFP page (See *FM01: Process Payments*). For Child Caring Institutions (CCI) and Group Homes, the percentage of federally reimbursable cost is based on two percentages. The first percentage is the individual maintenance percentage established for a provider on the Provider's Eligibility Cost page (See *PM02b: Private Provider Case File*). The second percentage is the percentage stored on Maintain Title IVE-FFP page. These two percentages applied to non-trust amount, and result in the portion of payment that is federally reimbursable. For a CCI, the State may also claim Title XIX money. The percentage is applied to State's portion of cost after the above calculation is completed. This percentage is also found on the Provider's Eligibility Cost page.

The results of these calculations may be viewed online for each individual payment can be viewed through the Payment Reimbursement History page. This page includes a break down of the amounts and benefits withdrawn from the trust account. Secondly, a break down of the State and Federal portions are shown. And thirdly, the number of days within each eligibility status is displayed, based on the payment begin date and end date. The Payment Reimbursement History page is accessed from the Payment Download page, where user can retrieve all payments that

meet a specified criterion entered into the page by the user.  The IV-E reimbursement claiming will occur quarterly, using both the Benefits Payments report and the Case Counts report.

## 1.2.   Pages

### 1.2.1.   Payment Download Page



Results Box displaying: Payment ID, Amount Requested, Service Type, Business ID, Provider Org, Check Number

### 1.2.2.

## Payment Download Page



Results Box displaying: Payment Begin Date, Payment End Date, Check Date, Case ID, Person ID, Basic Amt

**1.2.3.**

## Payment Download Page



Results Box displaying: Exceptional Amount, Supplemental Amount, Administrative Amount, Payee ID, Payment Status, Reimb. Rpt. Category

**1.2.4.**

### Payment Download Page



Results Box displaying: Title IV-E Eligibility

## 1.2.4.1. Page Overview

Page Navigation

The Payment Download page is accessed from the eWiSACWIS main menu by selecting Maintain>Financial Download>Payments.

Page Summary

The Payment Download page displays payment details from the Payment table for the attributes specified in the window header. Users may download the accounting attributes and payment data into a spreadsheet for additional analysis. This data is used to monitor fiscal performance, perform spending analysis, create forecasts and audit fiscal activities.

The Selection group box allows the user to specify search criteria for selecting records. The user is able to choose any combination of selection criteria, but must choose at least one. The Provider ID, Payee ID, Case ID and Person ID are user-entered text fields and will retrieve all payments with an exact match. A broader search is available through Service Type and

Reimbursement Category drop-down list fields, which will retrieve all payments that meet the value selected in the drop down. Additionally, the user may search by Check Number or Check Date Range. The user has the ability to retrieve payments that fall within a specified begin date range, and/or a specified end date range. If the Check Number field is used individually or with a combination of other search criteria fields, the user is restricted to 100 search results. If more are returned, a notification is given asking the user to limit the search.

The user may leave fields blank to select all payment records for that particular attribute. The user also has the option of narrowing the selection to target a highly specific subject area or group of like costs by specifying a value, or widening the selection with a blank field to allow broader retrieval of payment data.

After the user has selected all criteria necessary and hits the Search button, the Results group box displays detailed fiscal information for all payment records returned. Using the lower scroll bar the user can view additional attributes of each payment. Users may then download the selected information to a spreadsheet by selecting the Export Payment List option under the Options drop-down field. The user is able to name the file, which is then saved in the eWiSACWIS folder.

### 1.2.4.2. Page Information

**Box**     Selection

**Fields**   Provider Org ID:     User entered. Indicates the Provider Organization ID for which a payment needs to be downloaded from the Payment table. A blank will perform a search for all provider organizations within a payment line. Search is performed on id_prvd_org of PAYMENT table.

Payee ID:     User entered. Indicates the Payee ID for which a payment needs to be downloaded from the Payment table. A blank will perform a search for all payees within an payment line. Search is performed on the id_payee of the PAYMENT table.

Case ID:     User entered. Indicates the Case ID for which a payment needs to be downloaded from the Payment table. A blank will perform a search for all cases within an payment line. Search is performed on the id_case of the PAYMENT table.

Participant ID:     User entered. Indicates the Person ID for which a payment needs to be downloaded from the Payment table. A blank will perform a search for all persons within a payment line. Search is performed on the id_prsn of the PAYMENT table.

| | |
|---|---|
| Service Type: | User selected drop-down that lists all Service Types. Indicates the services for which an account needs to be downloaded from the Payment table. A blank will perform a search for all services within an account line. Search is performed on the cd_srvc of the PAYMENT table, which is mapped to cd_srvc of SERVICE_TYPE table where text descriptions are located. |
| Reimbursement Category: | User selected drop-down that displays the four placement settings of Group Home, CCI, Kinship Care, Foster Home. Search is performed on cd_plcm_setng on PAYMENT table. On CODE_DESC.id_grp = PLCMSTNG, where Group Home = cd_cdesc_type = 1, CCI = cd_cdesc_type = 2, Kinship Care = cd_desc_type = 4, Foster Home = cd_cdesc_type = 5, Adoption = 6 |
| Check Number: | User entered. User indicates the check number for which a payment needs to be downloaded from the payment table. Search is performed on PAYMENT.id_chck_nmbr. Using this field individually or in combination limits the search results to 100. |
| Check Date Range (Begin and End): | User entered. Check date range to use as record selection criteria from Payment table. Search is performed on PAYMENT.dt_chck |
| Begin Date of Payment Range (Begin and End): | User entered. Begin date of payment range to use as record selection criteria from Payment table. Search is performed on PAYMENT.dt_pmnt_begin. |
| End Date of Payment Range (Begin and End): | User entered. End date of payment range to use as record selection criteria from Payment table. Search is performed on PAYMENT.dt_pmnt_end. |

**Box**   Results

**Fields**

| | |
|---|---|
| Payment ID: | The system derived Payment ID of the associated payment. Retrieved from PAYMENT.id_pmnt. |

Amount Requested: The system derived amount of the associated payment. Retrieved from PAYMENT.am_rqst.

Service Type: The system derived service type of the associated payment. Retrieved from PAYMENT.cd_srvc, which is mapped to cd_srvc of SERVICE_TYPE table where text descriptions are located.

Business ID: The system derived Business ID of the associated payment. Retrieved from PAYMENT.id_bsns.

Provider Org ID: The system derived Provider Organization ID of the associated payment. Retrieved from PAYMENT.id_prvd_org.

Check Number: The system derived Check Number of the associated payment. Retrieved from PAYMENT.id_chck_nmbr.

Payment Begin Date: The system derived begin date of the associated payment. Retrieved from PAYMENT.dt_pmnt_begin.

Payment End Date: The system derived end date of the associated payment. Retrieved from PAYMENT.dt_pmnt_end.

Check Date: The system derived check date of the associated payment. Retrieved from PAYMENT.dt_chck.

Case ID: The system derived Case ID of the associated payment. Retrieved from PAYMENT.id_case.

Person ID: The system derived Person ID of the associated payment. Retrieved from PAYMENT.id_prsn.

Basic Amount: The system derived basic amount of the associated payment. Retrieved from PAYMENT.am_basic.

Exceptional Amount: The system derived exceptional amount of the associated payment. Retrieved from PAYMENT.am_excpt.

Supplemental
Amount: The system derived supplemental amount of the associated payment. Retrieved from PAYMENT.am_suppl.

Administrative
Amount: The system derived administrative amount of the associated payment. Retrieved from PAYMENT.am_admin.

Payee ID: The system derived Payee ID of the associated payment. Retrieved from PAYMENT.id_payee

Payment Status: The system derived payment status of the associated payment as either Cancel or Rescheduled or Blank. Retrieved from PAYMENT.cd_pmnt_stat.

Reimb. Rpt.
Category: The system derived reimbursement reporting category of the associated payment. Retrieved from cd_plcm_setng on PAYMENT table. On CODE_DESC.id_grp = PLCMSTNG, where Group Home = cd_cdesc_type = 1, CCI = cd_cdesc_type = 2, Kinship Care = cd_desc_type = 4, Foster Home = cd_cdesc_type = 5, Adoption = cd_cdesc_type = 6.

Title IV-E
Eligibility: The system derived Title IV-E Eligibility of the associated payment. Retrieved from the SERVICE_TYPE.cd_ive_12, where 1 = Non-Title IV-E (Regular), 2 = Title IV-E (Regular), 3 = Title IV-E (50/50).

Payment
Description: The system derived description of the associated payment. Retrieved from the PAYMENT. Tx_pmnt_desc

**Options**

Export Payment List    Opens the Save As pop-up page that provides the user with an opportunity to transfer this data to a spreadsheet. The exported spreadsheet will not include the user-entered Payment

Description field.

**Links**

> History             Opens the Payment Download History page. Enabled only when the Results box has been populated.

**Buttons**

> Search:          Search the Payment table for records meeting the selection criteria specified by the user.

> Close:          Closes Payment Download page.

> Clear:          Sets all the Selection criteria back to null or empty.

### 1.2.4.3. Background Processing

- After the user presses the Search button, the following occurs:

  - Using the accounting attributes entered by the user, retrieve the associated payment lines containing these attributes from the eWiSACWIS database Payment table. For the search attributes which the user has specified a wildcard value (blank), retrieve all of the records for the attribute.

  - Clear the data residing in the Results group box from a previous search.

  - Fill the Results data elements based on the outcome of the current account selection process.

  - Verifies that if the Check Number field is part of the search criteria, no more then 100 search results can be returned.

### 1.2.4.4. Save Processing

None

### 1.2.4.5. Page CRUD Matrix

| Table Name | CRUD |
|---|---|
| PAYMENT | R |
| SERVICE_TYPE | R |
| CODE_DESC | R |

## 1.2.5.  Payment Reimbursement History Page



### 1.2.5.1. Page Overview

Page Navigation

The Payment Reimbursement History page is accessed by clicking on the History hyperlink for a specific payment entry on the Payment Download page.

Page Summary

There are four components to this page.  The header of the page shows basic description information of when the payments was issued, for whom and on behalf of which case and child.

The Amount located in the Basic Information box can be derived from two equations.

1)  Amount (located on Header) = Payment Amount – Administrative Amount

2)  Amount (located on Header) = Total of Trust Account Draw Downs +
                                Federal Amount +
                                State Amount

The Trust Account Draw Downs box on this page shows a break down of what benefits were withdrawn from the child's trust account toward this payment. This page will display each benefit type used and how much of each benefit was withdrawn. These withdrawals will be shown as negative amounts. The date in this box shows the date on which the funds were withdrawn from the Trust Account.

The Title IV-E Reimbursement History box displays the break down between Federal and State components. The Maintenance Amount is shown as the amount of the payment that is Federally Title IV-E claimable and the remaining amount that is State's obligation. Also displayed are the percentages applied in the calculation of the Federal and State portion. The Federal Financial Participation (FFP) percentage is retrieved from the Maintain Title IVE-FFP page, and will only be displayed on the Federal row. The Maintenance percentage is retrieved from the Provider's Eligibility Cost page, and will only be displayed on the Federal row. The Title XIX Amount is based on the Title XIX percentage, which is retrieved from the Provider's Eligibility Cost page. The calculations are described below.

Once Trust Account money has been taken out against this payment, a Non-Trust Amount is established.

1) Non-Trust Amount (not displayed) = Amount (located on Header) – Trust Account Draw Down Total.

The Federal Amount located in the Title IV-E Reimbursement History box is calculated using one of three methods. The first two equations are applied only if the payment has at least one day of a status of Eligible and Reimbursable or Federal as indicated in the Days of Care History box. The first method is applied for out-of-home care eligibility payments that have a service type of Title IV-E (Regular). The second method is applied for adoption assistance eligibility payments that also have a service type of Title IV-E (Regular). The third method is applied for adoption assistance eligibility payments that have a service type of Title IV-E (50/50). This federal assistance is applied regardless of child's eligibility.

1) <u>IV-E Federal Amount</u>

$$IV\text{-}E\ Federal\ Amount = Non\text{-}Trust\ Amount \times \\ IV\text{-}E\ Prorated\ Percentage \times \\ Maintenance\ Percentage \times \\ FFP\ Percentage$$

where,

$$IV\text{-}E\ Prorated\ Percentage = \frac{Elig\ Reimb\ Days}{Elig\ Reimb\ Days + Elig\ Nonreimb\ Days + Ineligible\ Days + Pending\ Days}$$

2) <u>Adoption Assistance Federal Amount</u>

$$Adoption\ Federal\ Amount = Non\text{-}Trust\ Amount \times \\ Adoption\ Prorated\ Percentage \times \\ FFP\ Percentage$$

where,

$$\text{Adopton Prorated Percentage} = \frac{\text{Federal Days}}{\text{Federal Days} + \text{State Days} + \text{Pending Days}}$$

3) <u>Adoption Assistance 50/50 Federal Amount</u>

$$\text{Adoption Federal Amount} = \text{Non - Trust Amount} \times 50 \text{ Percent}$$

The State Amount located in the Title IV-E Reimbursement History box is derived from the following equation.

- State Amount = Non-Trust Amount - Federal Amount

However, the actual State obligation would be calculated as follows, because the Title XIX amount is considered part of the State Amount.

- State Obligation Amount (Not Displayed) = State Amount – Title XIX Amount

The Title XIX amount is derived from the following equation and applied only if a payment was for a Child Caring Institution (CCI):

- Title XIX Amount = Non-Trust Amount X Title XIX percentage X FFP Percentage

The Maintenance Percentage, State Percentage and Title XIX Percentage mentioned above are based on the Provider's Eligibility Cost page (*PM02b: Private Provider Case File*). These percentages are only applied to Private Providers, and the following equation must be maintained.

- 100% (Not Displayed) = Maintenance Percentage + State Percentage (Displayed on Provider's Eligibility Cost page) + Title XIX Percentage

Home Providers only have the FFP percentage applied to the Non-Trust Amount, if the correct eligibility criteria are met.

If a payment does not have at least one day of a status of Eligible and Reimbursable or Federal as indicated in the Days of Care History box, then the State is responsible to pay any Non-Trust Amount.

The Days of Care History box displays the number of days each payment was earmarked within an eligibility status. Event II payments will not span more than one calendar month. Each payment will have a begin date and end date for the period in which services were rendered. Each day between begin date and end date must be ascribed the appropriate eligibility status that was record for that child in that period (See *FM03: Eligibility*). If there is a change in eligibility status mid-month, then a break down of days within each eligibility status will be displayed. The payments for out of home care services will fall under the eligibility statuses of: Eligible

and Reimbursable, Eligible and Non-Reimbursable, Ineligible and Pending. The payments for adoption services will fall under the eligibility statuses of: Federal, State and Pending.

### 1.2.5.2. Page Information

**Box**     Basic

**Fields**

Case Name:     System derived. The name of the case (last name, first name, middle initial). This field pre-fills from the payment record selected on the Payment Download page. Retrieved from CASE.nm_case_frst and CASE.nm_case_lst

Person Name:     System derived. The name of the person (last name, first name, middle initial). This field pre-fills from the payment record selected on the Payment Download page. Retrieved from PERSON.nm_lst, nm_frst, nm_mdl_intl.

Provider Name:     System derived. The name of the provider (provider name, provider first name). This field pre-fills from the payment record selected on the Payment Download page. Retrieved from CHECKS.tx_payee.

Payee Name:     System derived. The payee name of the provider. This field pre-fills from the payment record selected on the Payment Download page. Retrieved from PROVIDER_ORG.tx_payee.

Service Type:     System derived. The service type of the payment. This field pre-fills from the payment record selected on the Payment Download page. Retrieved from SERVICE_TYPE.tx_srvc.

Reimbursement
Ctgry:     System derived. The reimbursement category of the payment. This field pre-fills from the payment record selected on the Payment Download page. Retrieved from cd_plcm_setng on PAYMENT table. On CODE_DESC.id_grp = PLCMSTNG, where Group Home = cd_cdesc_type = 1, CCI = cd_cdesc_type = 2, Kinship Care = cd_desc_type = 4, Foster Home = cd_cdesc_type = 5, Adoption = cd_cdesc_type = 6.

Begin Date: System derived. The begin date of the payment. This field pre-fills from the payment record selected on the Payment Download page. Retrieved from PAYMENT.dt_pmnt_begin.

End Date: System derived. The end date of the payment. This field pre-fills from the payment record selected on the Payment Download page. Retrieved from PAYMENT.dt_pmnt_end.

Check Date: System derived. The check date of the payment. This field pre-fills from the payment record selected on the Payment Download page. Retrieved from PAYMENT.dt_chck.

Amount: System derived. The amount of the payment minus any administrative costs. Retrieved from PAYMENT.am_rqst – PAYMENT.am_admin.

**Box** Trust Account Draw Downs

**Fields**

Date: System derived. The transaction date of the Trust Account Ledger records associated with the payment record selected on the Payment Download page. See background processing.

Amount: System derived. The amount of the Trust Account Ledger records associated with the payment record selected on the Payment Download page. See background processing.

Benefit Type: System derived. The benefit type of the Trust Account Ledger records associated with the payment record selected on the Payment Download page. See background processing.

**Box** Title IV-E Reimbursement History

**Fields**

Date: System derived. The creation date of the Payment Reimbursement records associated with the payment record selected on the Payment Download page. Retrieved from PAYMENT_REIMBURSEMENT.ts_cr.

| | |
|---|---|
| Source: | System derived. The source (either Federal or State) of the Payment Reimbursement records associated with the payment record selected on the Payment Download page. Retrieved from PAYMENT_REIMBURSEMENT.cd_reimb_type, where Federal = 1 and State = 2. |
| Status: | System derived. The status (either Active or Inactive) of the Payment Reimbursement records associated with the payment record selected on the Payment Download page. Retrieved PAYMENT_REIMBURSEMENT.cd_stat, where Inactive = I and Active = A. |
| Maintenance Amount: | System derived. The maintenance amount of the Payment Reimbursement records associated with the payment record selected on the Payment Download page. Retrieved from PAYMENT_REIMBURSEMENT.am_reimb. |
| Title XIX Amount: | System derived. The Title XIX amount of the Payment Reimbursement records associated with the payment record selected on the Payment Download page. Retrieved from PAYMENT_REIMBURSEMENT.am_titlexix. |
| FFP %: | System derived. The Federal Funding Participation percentage of the Payment Reimbursement records associated with the payment record selected on the Payment Download page. Retrieved from PAYMENT_REIMBURSEMENT.am_ffp_pct. |
| Maintenance %: | System derived. The Maintenance percentage of the Payment Reimbursement records associated with the payment record selected on the Payment Download page. Retrieved from PAYMENT_REIMBURSEMENT.am_maint_pct. |
| Title XIX %: | System derived. The Title XIX percentage of the Payment Reimbursement records associated with the payment record selected on the Payment Download page. Retrieved from PAYMENT_REIMBURSEMENT.am_titlexix_pct. |
| Pre 2yr Rpt Dt | The date on which the payment record was processed by FM04a08 – Pre Two Year Deduction report. This field is for |

auditing purposes to reflect the fact that money has been identified for refund, in support of the Federal mandate of reimbursing over-claimed Federal money. The date is updated by the batch report and is not user modifiable. PAYMENT_REIMBURSEMENT.dt_pre_2yr.

**Box:** **Days of Care History**

**Fields:**

Date: System derived. The creation date of the Days of Care records associated with the payment record selected on the Payment Download page. Retrieved from PAYMENT_ELIG_DAYS.ts_cr.

Status: System derived. The status of the Days of Care History record (either Active or Inactive). Retrieved from PAYMENT_ELIG_DAYS.cd_stat, where Inactive = I and Active = A.

Eligible –
Reimbursable: System derived. The number of days in the payment period that the eligibility status of the person associated with the payment was eligible and reimbursable. Retrieved from PAYMENT_ELIG_DAYS.qt_elig_reimb.

Eligible –
Non-reimbursable: System derived. The number of days in the payment period that the eligibility status of the person associated with the payment was eligible and non-reimbursable. Retrieved from PAYMENT_ELIG_DAYS.qt_elig_nonreimb.

Ineligible: System derived. The number of days in the payment period that the eligibility status of the person associated with the payment was ineligible. Retrieved from PAYMENT_ELIG_DAYS.qt_inelig.

SSI: System derived. The number of days in the payment period that the eligibility status of the person associated with the payment was SSI. Retrieved from PAYMENT_ELIG_DAYS.qt_ssi.

Pending: System derived. The number of days in the payment period that the eligibility status of the person associated with the payment

was pending.  Retrieved from PAYMENT_ELIG_DAYS. qt_pending.

Federal:  System derived.  The number of days in the payment period that the eligibility status of the person was earmarked for Federal Financial Participation. Retrieved from PAYMENT_ELIG_ DAYS.qt_federal.

State:  System derived.  The number of days in the payment period that the person associated with the payment was not earmarked for Federal Financial Participation and is the obligation of the State. Retrieved from PAYMENT_ELIG_DAYS.qt_state.

### 1.2.5.3.  Background Processing

- For the Trust Account Draw Downs Group Box, retrieve all TRUST_ACCOUNT_LEDGER records WHERE TRUST_ACCOUNT_LEDGER.id_pmnt = PAYMENT.id_pmnt AND TRUST_ACCOUNT_LEDGER.am_trans < 0. Sort the selected rows with the most recent dt_trans at the top

- For the Title IV-E Reimbursement History Group Box, retrieve all PAYMENT_REIMBURSEMENT records WHERE PAYMENT_REIMBURSEMENT.id_pmnt = PAYMENT.id_pmnt. Sort the selected rows with the most recent ts_cr at the top.

- For the Days of Care History Group Box, retrieve all PAYMENT_ELIG_DAYS records WHERE PAYMENT_ELIG_DAYS.id_pmnt = PAYMENT.id_pmnt. Sort the selected rows with the most recent ts_cr at the top.

### 1.2.5.4.  Save Processing

None

### 1.2.5.5. Page CRUD Matrix

| Table Name | CRUD |
|---|---|
| PAYMENT | R |
| PERSON | R |
| CASE | R |
| CODE_DESC | R |
| PROVIDER_ORG | R |

| PAYMENT_REIMBURSEMENT | R |
|---|---|
| PAYMENT_ELIG_DAYS | R |
| TRUST_ACCOUT_LEDGER | R |

## 1.2.6.  Eligibility-Based Days of Care Page



### 1.2.6.1.Page Overview

Page Navigation:

The Eligibility-Based Days of Care page is opened in a View Only mode and users will not be able to edit any information on this page.  The page will be available from the main Outliner > Maintain menu.  Access to the Eligibility-Based Days of Care page will be maintained by the overall job class level.

Page Summary:

The Eligibility-Based Days of Care page will allow workers and auditors look at the days of care broken down by eligibility and reimbursibility status, by child.  Using this page, users are

able to drill down historical days of care information provided by the Eligibility-Based Days of Care report at the level of a child and eligibility. The user can search for information by child or person and by date range for the case or person. Once the user views this information, they will be able to save the information into a spreadsheet through the Options menu. After saving the data, the user will be returned to the Eligibility-Based Days of Care page.

The page is divided into two group boxes. The first group box is used by the user to specify selection criteria. The lower part of the page will display the search results based on the selection criteria specified. The Options drop down will allow the user to download the information displayed in the Eligibility Based Days of Care Results group box.

### 1.2.6.2.Page Information

**Box**      Selection Criteria Group Box

**Fields**

| | | |
|---|---|---|
| | Person ID: | Specifies a unique person ID (WISACWIS generated person ID); can be used as the sole search criterion; user-entered field; enabled. Mandatory. |
| | Case ID: | Specifies a unique case ID (WISACWIS generated); user-entered field; enabled. |
| | Date From: | Begin Date from which time period eligibility information is required. User-entered field in MM/DD/CCYY format; enabled. Mandatory. |
| | Date To: | End date up to which time period eligibility information is required. User-entered field in MM/DD/CCYY format; enabled. |
| **Button** | Search: | Standard Search processing, using the specified criteria. |
| | Clear: | Standard Clear search criteria specified. |

**Box**      Eligibility Based Days of Care Results Group Box

**Fields**

| | | |
|---|---|---|
| | Eligibility ID: | Eligibility ID for the particular record that is associated with the placement and eligibility information being reported for the part. |
| | Person Name: | Display the Person First and Last Name based on the Person ID displayed. |

|  | Case ID: | Case ID associated with the person and eligibility ID for which Eligibility-Based Days of Care information is brought up. |
|---|---|---|
|  | County: | Name of the County in which the child's eligibility status is associated with. |
|  | Period Begin Date: | Begin Date of the monthly reporting period associated with the days of care being reported. |
|  | Period End Date: | End Date of the monthly reporting period associated with the days of care being reported. |
|  | Placement Begin Date: | Begin date of the placement associated with the eligibility ID and the Period Begin Date. |
|  | Service Type: | Service Type associated with the placement. |
|  | Days– ER: | Number of days within the monthly period and the eligibility instance that the child had a status of Eligible and Reimbursable. |
|  | Days– ENR: | Number of days within the monthly period and eligibility instance that the child had a status of Eligible and Non-Reimbursable. |
|  | Days – Elig-SSI: | Number of days within the monthly period and eligibility instance that the child had a status of Eligible, receiving SSI. |
|  | Days – Ineligible: | Number of days within the monthly period and the eligibility instance that the child had a status of Ineligible. |
|  | Days – Pending: | Number of days within the monthly period and eligibility instance that the child had a status of Pending. |
|  | Status of Record: | Active or Inactive status of the record being displayed. |
|  | Time Created: | Time at which the particular monthly days of care record was created in the Days_of_Care_Period table. |
| **Button** | Close: | Standard Close processing. |
| **Options Menu Drop Down** | Export Days of Care List: | Allows the user to export search results into a spreadsheet. |

### 1.2.6.3. Background Processing

- Person ID is a mandatory Search criterion.

- Date From is mandatory Search criterion.

- All fields default to null values when the page is initially accessed.

- The Date To cannot be prior to the Date From field.

- For the user-specified Date From, retrieve those records where user-specified Date From <= DAYS_OF_CARE_PERIOD.dt_begin.

- If the user specifies a Date To, retrieve those records where user-specified Date To => DAYS_OF_CARE_PERIOD.dt_begin.

- If no Date To is specified, all records until the system date (current date) are retrieved.

- If only Person ID and Date From are specified, all Days of Care records for the person, irrespective of case, is retrieved. user-specified Person ID = DAYS_OF_CARE_PERIOD.id_prsn.

- If the user presses the Clear button, all search criteria are cleared out.

- Records are retrieved after the Search button is clicked.

- Fields are pre-filled from values driven off the Days_of_Care_Period table, based on the criteria specified.

- Data is sorted in the following order:

  - Case ID, Eligibility ID, Period Begin Date (descending), Cd_stat = 'A', Cd_stat = 'I'

- Multiple records for the same eligibility instance and the same placement can be displayed from the table. Records are reported based on the Period Begin Date, a monthly division of days of care information.

- If DAYS_OF_CARE_PERIOD.id_epsd = null, Placement Begin Date and Service Type fields are null. This indicates a gap in placement.

- The Service Type field is obtained from the EPISODE table. Display SERVICE_TYPE.tx_srvc where SERVICE_TYPE.cd_srvc = EPISODE.cd_srvc and where EPISODE.id_epsd = DAYS_OF_CARE_PERIOD.id_epsd.

- The placement begin date field is obtained from the EPISODE table. Display EPISODE.dt_bgn where EPISODE.id_epsd = DAYS_OF_CARE_PERIOD.id_epsd.

- The Eligibility-Based Days of Care page is brought up as a View Only page.

- If no data is found for the specified Search criteria, a message stating "No Days of Care information found for specified criteria" will be displayed.

- By choosing 'Export Days of Care List' in the Options Menu Drop down (under Actions), the user will be able to save the records displayed onto a spreadsheet, in the eWISACWIS folder.

o The column titles on the spreadsheet will match the field headings in the page display.

o Results displayed on the Eligibility-Based Days of Care page are downloaded into the spreadsheet.

### 1.2.6.4.      Save Processing

- None.

### 1.2.6.5. Page CRUD Matrix

| Table Name | Create | Read | Update | Delete |
|---|---|---|---|---|
| EPISODE | | X | | |
| DAYS_OF_CARE_PERIOD | | X | | |
| SERVICE_TYPE | | X | | |

## 1.3.  Inventories

### 1.3.1.  Table Descriptions

| Table Name | Description |
| --- | --- |
| CASE | A CASE is the focus of every human services business activity. A case is created as a result of intake work and can be one of three types: individual (which includes TPR and Adoption cases), family, and DHFS provider or other provider. The CASE table contains case relations, address and approval information. Processes of SM04 (Maintain Case) and SM05 (Close Case) create and maintain case records. |
| CODE_DESC | The CODE DESC table is used to house the description fields for most of the reference data used in eWiSACWIS. |
| PAYMENT | The PAYMENT table stores detailed information about payments made to providers including but not limited to service begin date, service end date, payment amount, provider, child, case, and payee. Processes of FM01 (Process Payments) maintain and update this information. |
| PAYMENT_ELIG_DAYS | The PAYMENT_ELIG_DAYS table stores the number of days each payment was earmarked within an eligibility status. Processes of FM04a (Reimbursement) maintain and update this information. |
| PAYMENT_REIMBURSEMENT | The PAYMENT_REIMBURSEMENT table contains a history of system calculated reimbursement claim amounts. Processes of FM04a (Reimbursement) maintain and update this information. |
| PERSON | This PERSON table maintains information that identifies an individual known to DHFS or the county child welfare division such as name, date of birth, social security number, race, sex, etc. A PERSON can be a WORKER, REPORT PART, REFERRAL PART, CASE PART or PROVIDER PART. Primary search processing is centered |

|  |  |
|---|---|
|  | around this data. Processes of CM01 (Manage Person) build and maintain this information. |
| PROVIDER_ORG | This table maintains information pertaining to a PROVIDER ORG, facility or vendor. Data includes name and address information, placement preferences and home condition description. Records in PROVIDER ORG table are created in PM02a/b (Maintain Home/ Private Provider) and updated in PM04a (Licensing Home Provider). |
| SERVICE_TYPE | SERVICE TYPE table maintains information associated with any defined category of provider service offered by DHFS or the county organization including training requirements, standard rate (across all provider organizations) and license requirements etc., associated with providers. Processes of PM01 (Maintain Services) and eWiSACWIS table maintenance build and maintain this data. |
| TRUST_ACCOUNT_LEDGER | The TRUST_ACCOUNT_LEDGER table is used by the Trust Account Manager to maintain a complete history of all debit and credit transactions for a particular trust account. Processes within FM07 (Trust Account) maintain this information. |

## 1.3.2.  Reference Data

### 1.3.2.1. Drop Downs

**Field Name:**   Service Code
**Table Name:**   SERVICE_TYPE

**Field Name:**   Reimbursement Category
**Table Name:**   CODE_DESC
**Group ID:**   PLCMSTNG

### 1.3.2.2. List Boxes

None.

## 1.3.3.  Automated Messages

None

**1.3.4. Checklists**

None

**1.3.5. Ticklers**

None

**1.3.6. Notifications**

None

**1.3.7. Text Templates**

None

**1.3.8. Reports**

1. Benefits Payments Report

2. Case Counts Report

3. Payment Based Days of Care Report

4. Overpayments with Trust Accounts Report

5. Title IV-E Audit Report

6. Eligibility-based Days of Care Report

7. Pre-Two Year Deductions Report

**1.3.9. Triggers**

| Txn Code | Description |
|---|---|
| 7000 | Approval of initial out of home placement. This trigger is created if an initial out of home placement is  service rate change occurs with the effective date of the change (DT_EFCT_STRT on SERVICE_RATE/PRVD_SRVC_RATE Tables) being earlier than the last payment date for that particular service (DT_PMNT_LST on the SERVICE_TYPE Table). This trigger will be used by FM01 - Calculate Ongoing Amounts Batch to create retroactive payments for those provider who have already been paid although the rate change was effective earlier than their payment date. |

### 1.4. Batch Programs

### 1.4.1. Trust Funds Draw-down Batch Program

**Program Name:**     b-fm04a-trust-draw-down

**Process Summary:**     This batch program takes all current payments that have an existing Non Trust Amount (see section 1.2.2.1 for further details on this amount), and looks to see if there is any money available in the child's trust account. The batch will withdraw any available money and create report for the Trust Account Manager that will indicate the total amount reimbursable to the State for that month.   In order for the Trust Account benefits (See *FM07: Trust Accounts)* to be selected the following conditions must apply.

1) Payment Begin Date must be equal to or greater than the Effective Date of the Benefit

2) If the benefit type is specified for Current Month expenses only, then the Payment must fall in current month in which the batch is run.

3) If the benefit type is specified for Prior Month expenses only, then the Payment must fall in the prior month in which the batch is run.

4) If the benefit type is specified for All Outstanding expenses, then the Payment must only meet the criteria (1)

Additionally, the Trust Account Draw Down batch will always withdraw any existing interest record in the ledger.  However, any money that is record as Personal Needs in the ledger will not be withdrawn by this batch.

**Frequency:**     Monthly.

**Dependencies:**     Successful completion of monthly trust account deposits and DOA-to-WiSACWIS Checks Printed batch and Identify FFP/Non-FFP Payments batch programs.

**Input Parameters:**     mmddccyy

**Input Files:**     cycle-date-override, fm04a-trust-draw-down-parm

**Output Files:** Sum of all withdrawals output file (See Item 7 for description)

**Database Tables:** PAYMENT RU

TRUST_ACCOUNT_LEDGER CRU

**Process Description:**

1. The batch will now update new withdrawal entries, so that am_rem is appropriately reduced on prior deposit entries.

    1.1. Sort TRUST_ACCOUNT_LEDGER by dt_trans

    1.2. If fl_adjstd = N, and cd_trans_typ = 2 (withdrawal) then look up and retrieve first row where cd_bnft_typ = cd_bnft_typ (of this withdrawal record) and id_trust_acct = id_trust_acct (of this withdrawal record) and am_rem > 0

    1.3. Verify that TRUST_ACCOUNT_LEDGER.am_trans (of withdrawal) < sum(TRUST_ACCOUNT_LEDGER.am_rem) for cd_bnft_type.

        1.3.1. If true, proceed with pre-process.

        1.3.2. If false, go to next withdrawal.

    1.4. am_rem + am_trans (this is a negative amount) = WS-LEFT-OVER

    1.5. Loop

        1.5.1. If WS-LEFT-OVER = 0, THEN set am_rem = 0, THEN exit loop

        1.5.2. If WS-LEFT-OVER > 0, THEN SET AM_REM = WS-LEFT-OVER, THEN exit loop

        1.5.3. IF WS-LEFT-OVER < 0 THEN set am_rem = 0, AND then select next row where am_rem > 0, and take am_rem + WS-LEFT-OVER = WS-LEFT-OVER and GO TO top of Loop.

        1.5.4. Set fl_adjstd = Y (of this withdrawal record)

2. Select PAYMENT records.

    2.1. Select system disbursed payments. Select those records where PAYMENT.cd_pmnt_stat <> "C" AND PAYMENT.dt_chck is NOT NULL AND TRUST_ACCOUNT_LEDGER.id_prsn = PAYMENT.id_prsn AND PAYMENT.am_nontrust > 0 AND SERVICE_TYPE.fl_non_sys_disb = "N" AND SERVICE_TYPE.cd_ive_12 <> 3 AND SERVICE_TYPE.cd_ive_12 <> 4 (code value for 'exclude from claiming') for the PAYMENT.cd_srvc selected.

    2.2. Also, select non-disbursed payments. Select all **approved** PAYMENT records where TRUST_ACCOUNT_LEDGER.id_prsn = PAYMENT.id_prsn AND PAYMENT.cd_cnty_new = TRUST_ACCOUNT.cd_cnty AND

PAYMENT.cd_pmnt_stat <> "C" AND PAYMENT.am_nontrust > 0 AND
SERVICE_TYPE.fl_non_sys_disb = "Y" AND SERVICE_TYPE.cd_ive_12 <> 3 for
the PAYMENT.cd_srvc selected AND PAYMENT.cd_ffp = 2 or 3
SERVICE_TYPE.cd_sw_rpt_grp <> '999'.

2.3. Also, select non-disbursed payments. Select all **approved** PAYMENT records where
TRUST_ACCOUNT_LEDGER.id_prsn = PAYMENT.id_prsn AND
PAYMENT.cd_cnty_new = TRUST_ACCOUNT.cd_cnty AND
PAYMENT.cd_pmnt_stat <> "C" AND PAYMENT.am_nontrust > 0 AND
SERVICE_TYPE.fl_non_sys_disb = "Y" AND PAYMENT.cd_ffp <> 2 or 3 AND
SERVICE_TYPE.cd_ive_12 <> 3 for the PAYMENT.cd_srvc selected
SERVICE_TYPE.cd_sw_rpt_grp = '999'.

2.4. Sort the selected rows by PAYMENT.id_prsn.

2.5. Select the first payment record from this group. If there are multiple records, select the
records where PAYMENT.cd_ffp = "3" (i.e., Non-FFP) and then records with
PAYMENT.cd_ffp = "2" (i.e., FFP). If there are multiple records in each category, then
select the record with the least recent PAYMENT.dt_pmnt_end.

3. The next step is to find valid benefits that can be used toward each payment selected and
select TRUST_ACCOUNT_LEDGER records available for the month in which this payment
falls.

3.1. For the first PAYMENT.id_prsn and id_case and perform lookup to
TRUST_ACCOUNT and find row where TRUST_ACCOUNT.dt_closing = null and
BANK.cd_type = 1 (i.e. General Account) where TRUST_ACCOUNT.id_bank =
BANK.id_bank.

3.2. With TRUST_ACCOUNT.id_trust_acct lookup to BENEFIT.id_trust_acct and find
rows where BENEFIT.dt_effctv (month and year) =< PAYMENT.dt_pmnt_begin
(month and year) =< BENEFIT.dt_clsng (month and year). Verify that
sum(TRUST_ACCOUNT_LEDGER.am_trans>0) for the benefit type. If true, continue
processing payment with that benefit. If false, go to next benefit defined in BENEFIT
table.

3.3. Select rows on BENEFIT where cd_expns = 2 (current month benefit).  Lookup to
TRUST_ACCOUNT_LEDGER with id_trust_acct and BENEFIT.cd_bnft_typ =
TRUST_ACCOUNT_LEDGER.cd_bnft_typ and select rows on
TRUST_ACCOUNT_LEDGER where TRUST_ACCOUNT_LEDGER.am_rem > 0 and
month of dt_trans = month of dt_pmnt_begin.

3.4. Select rows on BENEFIT where cd_expns = 3 (prior month benefit).  Lookup to
TRUST_ACCOUNT_LEDGER  with id_trust_acct and BENEFIT.cd_bnft_typ =
TRUST_ACCOUNT_LEDGER.cd_bnft_typ and select rows on
TRUST_ACCOUNT_LEDGER where TRUST_ACCOUNT_LEDGER.am_rem > 0 and
month of dt_trans minus one month (make sure 01 – 1 = 12) = month of dt_pmnt_begin.

3.5. Select rows on BENEFIT where cd_expns = 1 (all outstanding).  Lookup to
TRUST_ACCOUNT_LEDGER  with id_trust_acct and BENEFIT.cd_bnft_typ =

TRUST_ACCOUNT_LEDGER.cd_bnft_typ and select rows on TRUST_ACCOUNT_LEDGER where TRUST_ACCOUNT_LEDGER.am_rem > 0

3.6. Also select TRUST_ACCOUNT_LEDGER rows where cd_bnft_typ = 10 (i.e. Interest) where cd_bnft_type = 10 does not exist in the BENEFIT table for the TRUST_ACCOUNT.id_trust_acct.

3.7. Once the rows have been selected place all rows cd_bnft_typ = 6 (i.e. SSI-Federal) last. For cd_bnft_type = 6, verify that BENEFIT.dt_effctv (month and year) =< PAYMENT.dt_pmnt_begin (month and year) =< BENEFIT.dt_clsng (month and year). Verify that sum(TRUST_ACCOUNT_LEDGER.am_trans>0) for the benefit type. If true, continue processing payment with that benefit. If false, go to next benefit defined in BENEFIT table. Next sort on dt_trans.

3.8. Calculate the Current Balance of the Trust Account by adding all TRUST_ACCOUNT_LEDGER.am_rem for these selected TRUST_ACCOUNT_LEDGER entries. Refer to this as "Current Balance"

4. Insert a TRUST_ACCOUNT_LEDGER record with the following attributes:

4.1. TRUST_ACCOUNT_LEDGER.id_case = PAYMENT.id_case

4.2. TRUST_ACCOUNT_LEDGER.id_prsn =PAYMENT.id_prsn

4.3. TRUST_ACCOUNT_LEDGER.id_ldgr = system generated unique ledger ID.

4.4. TRUST_ACCOUNT_LEDGER.id_chck_nmbr = NULL

4.5. TRUST_ACCOUNT_LEDGER.dt_trans = current system date

4.6. TRUST_ACCOUNT_LEDGER.cd_trans_typ = "2" (i.e., Withdrawal)

4.7. TRUST_ACCOUNT_LEDGER.am_trans =

    4.7.1. If PAYMENT.am_nontrust >= "Current Balance" >= TRUST_ACCOUNT_LEDGER.am_rem of the originally selected TRUST_ACCOUNT_LEDGER record, then

        4.7.1.1. Set TRUST_ACCOUNT_LEDGER.am_trans = - (TRUST_ACCOUNT_LEDGER.am_rem).

        4.7.1.2. Set TRUST_ACCOUNT_LEDGER.am_rem = 0

        4.7.1.3. Set PAYMENT.am_nontrust = (PAYMENT.am_nontrust - TRUST_ACCOUNT_LEDGER.am_rem)

        4.7.1.4. Set PAYMENT.fl_ive_calc = "Y" and continue to (4.8).

    4.7.2. If PAYMENT.am_nontrust >= TRUST_ACCOUNT_LEDGER.am_rem of the originally selected TRUST_ACCOUNT_LEDGER record >= "Current Balance", then

        4.7.2.1. Set TRUST_ACCOUNT_LEDGER.am_trans = - (Current Balance).

        4.7.2.2. Set TRUST_ACCOUNT_LEDGER.am_rem = (TRUST_ACCOUNT_LEDGER.am_rem – Current Balance)

4.7.2.3.    Set PAYMENT.am_nontrust = (PAYMENT.am_nontrust – Current Balance)

4.7.2.4.    Set PAYMENTfl_ive_calc = "Y" and continue to (4.8).

4.7.3.   If "Current Balance" >= PAYMENT.am_nontrust >= TRUST_ACCOUNT_LEDGER.am_rem of the originally selected TRUST_ACCOUNT_LEDGER record, then

4.7.3.1.    Set TRUST_ACCOUNT_LEDGER.am_trans = - (TRUST_ACCOUNT_LEDGER.am_rem).

4.7.3.2.    Set TRUST_ACCOUNT_LEDGER.am_rem = 0

4.7.3.3.    Set PAYMENT.am_nontrust = (PAYMENT.am_nontrust - TRUST_ACCOUNT_LEDGER.am_rem) and continue to (4.8).

4.7.4.   If "Current Balance" >= TRUST_ACCOUNT_LEDGER.am_rem of the originally selected TRUST_ACCOUNT_LEDGER record >= PAYMENT.am_nontrust, then

4.7.4.1.    Set TRUST_ACCOUNT_LEDGER.am_trans = - (PAYMENT.am_nontrust).

4.7.4.2.    Set TRUST_ACCOUNT_LEDGER.am_rem = (TRUST_ACCOUNT_LEDGER.am_rem – PAYMENT.am_nontrust)

4.7.4.3.    Set PAYMENT.am_nontrust = 0

4.7.4.4.    Set PAYMENTfl_ive_calc = "Y" and continue to (4.8).

4.7.5.   If TRUST_ACCOUNT_LEDGER.am_rem of the originally selected TRUST_ACCOUNT_LEDGER record >= PAYMENT.am_nontrust >= "Current Balance", then

4.7.5.1.    Set TRUST_ACCOUNT_LEDGER.am_trans = - (Current Balance).

4.7.5.2.    Set TRUST_ACCOUNT_LEDGER.am_rem = (TRUST_ACCOUNT_LEDGER.am_rem – Current Balance)

4.7.5.3.    Set PAYMENT.am_nontrust = (PAYMENT.am_nontrust – Current Balance)

4.7.5.4.    Set PAYMENTfl_ive_calc = "Y" and continue to (4.8).

4.7.6.   If TRUST_ACCOUNT_LEDGER.am_rem of the originally selected TRUST_ACCOUNT_LEDGER record >= "Current Balance" >= PAYMENT.am_nontrust, then

4.7.6.1.    Set TRUST_ACCOUNT_LEDGER.am_trans = - (PAYMENT.am_nontrust).

4.7.6.2.    Set TRUST_ACCOUNT_LEDGER.am_rem = (TRUST_ACCOUNT_LEDGER.am_rem – PAYMENT.am_nontrust)

4.7.6.3.    Set PAYMENT.am_nontrust = 0

4.7.6.4. Set PAYMENTfl_ive_calc = "Y" and continue to (4.8).

4.8. TRUST_ACCOUNT_LEDGER.cd_detl = the code value for "Reimbursement."

4.9. TRUST_ACCOUNT_LEDGER.cd_bnft_typ = the cd_bnft_typ of the originally selected TRUST_ACCOUNT_LEDGER record.

4.10. TRUST_ACCOUNT_LEDGER.tx_desc = NULL

4.11. TRUST_ACCOUNT_LEDGER.id_cr = Batch ID

4.12. TRUST_ACCOUNT_LEDGER.ts_cr = current system date

4.13. TRUST_ACCOUNT_LEDGER.id_up = Batch ID

4.14. TRUST_ACCOUNT_LEDGER.ts_up = current system date

4.15. TRUST_ACCOUNT_LEDGER.id_trust_acct = the id_trust_acct of the originally selected TRUST_ACCOUNT_LEDGER record.

4.16. TRUST_ACCOUNT_LEDGER.id_pmnt = PAYMENT.id_pmnt

4.17. TRUST_ACCOUNT_LEDGER.am_rem = 0.

5. Verify that sum(TRUST_ACCOUNT_LEDGER.am_trans)>=0 for the benefit type . Include in the sum the value in the TRUST_ACCOUNT_LEDGER.am_trans of the newly inserted record. If true, insert the record. If false, then select the next TRUST_ACCOUNT_LEDGER record that was selected in STEP 3 and continue from STEP 4.

6. If the TRUST_ACCOUNT_LEDGER.am_rem = 0 for the selected record, then select the next TRUST_ACCOUNT_LEDGER record that was selected in STEP 3 and continue from STEP 4. Perform this until PAYMENT.am_non_trust = 0, then get next id_pmnt of the PAYMENT rows select in STEP 2 for this id_prsn

   6.1. If the next id_pmnt falls in a different calendar month, then return to STEP 3 and reselect TRUST_ACCOUNT_LEDGER records and recalculate balance

   6.2. If the next id_pmnt falls in the same calendar month, then return to STEP 4.

7. If the Current Balance = 0, then select the next PAYMENT record that for this id_prsn that falls in a different calendar month and continue with STEP 3. If no PAYMENT record for this id_prsn, then select next PAYMENT row of the next id_prsn.

8. Also, each time a TRUST_ACOCUNT_LEDGER record is inserted, use a working storage to maintain the sum of all am_trans amounts entered. This sum will be the total of all withdrawals made.

9. Display this sum total in a (.txt) output file which shows the date the job ran and the "Total Amount of Withdrawals" = $(the calculated sum)

### 1.4.2. <u>Identify FFP/Non-FFP Payments Batch Program</u>

**Program Name:**    b-fm04a-identify-FFP

**Process Summary:**    This batch program processes the Title IV-E Eligibility for children who are in out-of-home-care and process Adoption Eligibility for children who are placed with in-home-care services.   The batch then makes two important decisions.  Firstly, the batch will decide if a payment is considered FFP or Non-FFP.  A payment is consider FFP if, and only if, at least one day of the payment falls in period when the eligibility status was Eligible and Reimbursable (out-of-home care), or the payment falls in a period when the status was Federal (in-home-care). Secondly, for each payment the batch will calculate the number of days earmarked for a particular eligibility status.  Event II payments each have a begin date and end date that fall within the same calendar month.  Each day must be allocated to the eligibility that had been determined for that child within that period.

**Frequency:**    Monthly.

**Dependencies:**    Successful completion of DOA to WiSACWIS Check Printed batch processing.

**Input Parameters:**    From Date.

**Input Files:**    cycle-date-override

**Output Files:**    None

**Database Tables:**

| | |
|---|---|
| PAYMENT | RU |
| ELIGIBILITY | R |
| ELIG_REDET | R |
| ADOPTION_ELIGIBILITY | R |
| ADOPTION_ELIG_REDET | R |
| SERVICE_TYPE | R |

PAYMENT_ELIG_DAYS          CU

**Process Description:**

The following batch program processes payments that have been disbursed since the From Date as recorded on the Input Parameter card. Based on the latest Title IV-E Eligibility information for the child, the batch program whether a particular payment is Title IV-E reimbursable (FFP) or not (Non-FFP). It also calculates the days of care for each eligibility and reimbursability status.

The first step is to process the eligibility rows entered in ELIGIBILITY and ELIG_REDET tables and select only those rows that actually change the status of a child. For example there may be two eligibility rows in ELIGIBILITY and ELIG_REDET tables with the following attributes:

| dt_eff = 3/15/1999 | dt_cmplt = 4/1/1999 | Status = Eligible and Reimbursable |
|---|---|---|
| dt_eff = 3/20/1999 | dt_cmplt = 3/25/1999 | Status = Eligible but Not-reimbursable |

In this instance, based on the effective dates, it may look like this child is Eligible and Reimbursable between 3/15 and 3/20 and becomes Eligible but Non-reimbursable after 3/20. However, since the Eligible and Reimbursable status was entered more recently than the Eligible but Non-reimbursable status, it overwrites this status. Therefore, this child is actually Eligible and Reimbursable after 3/15 and the eligibility record with the status Eligible but Non-reimbursable does not affect the overall eligibility or reimbursability of the child.

The above mentioned example is simplified version of the logic that will dictate the selection of "relevant" eligibility rows. The logic described below will need to be followed to select these relevant eligibility rows.

1. Select payments that can potentially be claimed from Title IV-E funds.

   1.1. Select system disbursed (+) payments. Select PAYMENT rows WHERE PAYMENT.cd_pmnt_stat <> "C" AND PAYMENT.am_nontrust >0 AND PAYMENT.fl_ffp_calc = "Y" AND PAYMENT.dt_chck >= From date entered on the input parameter AND SERVICE_TYPE.fl_non_sys_disb = "N" AND SERVICE_TYPE.cd_ive_12 <> 4 (code value for 'exclude from claiming') for the PAYMENT.cd_srvc selected.

   1.2. Select non-system disbursed (+) payments. Select **approved** PAYMENT rows WHERE PAYMENT.cd_pmnt_stat <> "C" AND PAYMENT.am_nontrust >0 AND PAYMENT.fl_ffp_calc = "Y" AND PAYMENT.dt_pmnt_end >= From date entered on the input parameter AND AND SERVICE_TYPE.fl_non_sys_disb = "Y" for the

PAYMENT.cd_srvc selected.  Do not select payments where SERVICE_TYPE.cd_sw_rpt_grp = '999'.

1.3. Select (-) payments. Select **approved** PAYMENT rows WHERE PAYMENT.am_rqst <0 AND PAYMENT. fl_ffp_calc = "Y" AND PAYMENT.dt_pmnt_end >= From date entered on the input parameter and . Do not select payments where SERVICE_TYPE.cd_sw_rpt_grp = '999' or SERVICE_TYPE.cd_ive_12 <> 4 (code value for 'exclude from claiming').

2.  Select the first PAYMENT record. If SERVICE_TYPE.cd_ive_12 = "1" (i.e., Non-Title IV-E) for the PAYMENT.cd_srvc selected, then set PAYMENT.cd_ffp = "3" (i.e., Non-FFP). Calculate dt_pmnt_end minus dt_pmnt_bgn +1.  Using id_prsn and id_case lookup to ELIGIBILITY (or ELIG_REDET), and if found insert result in to PAYMENT_ELIG_DAYS.qt_inelig. If not found use id_prsn and id_case lookup to ADOPTION_ELIG (or ADOPTION_ELIG_REDET), and if found insert result in to PAYMENT_ELIG_DAYS.qt_state.   If not found, insert result into qt_pending.  If SERVICE_TYPE.cd_ive_12 = "3" (i.e., Title IV-E (50/50)), or SERVICE_TYPE.cd_ive_12 = "2" (i.e., Title IV-E Regular), then continue from (3).

3.  The next step is to process those payments that are potentially Title IV-E reimbursable. The batch process first processes those payments that are made for children in adoption. The Title IV-E eligibility records for those children are recorded in the ADOPTION_ELIG and ADOPTION_ELIG_REDET tables. Therefore, select all ADOPTION_ELIG and ADOPTION_ELIG_REDET records WHERE PAYMENT.id_case = ADOPTION_ELIG.id_case (or ADOPTION_ELIG_REDET.id_case) AND PAYMENT.id_prsn = ADOPTION_ELIG (or ADOPTION_ELIG_REDET.id_prsn) AND ADOPTION_ELIG.dt_cmplt (or ADOPTION_ELIG_REDET.dt_cmplt) is NOT NULL AND ADOPTION_ELIG.dt_eff_elig (or ADOPTION_ELIG_REDET.dt_eff_elig) <= PAYMENT.dt_pmnt_end. If there is at least one record selected, then continue from (3.1). Once, all Adoption Eligibility records are processed, skip (4) and continue from (5). If there are no records in these tables for the selected payment, then the batch process will assume that the child is in foster care and will continue from (4) to process regular Title IV-E Eligibility determination records.

3.1. If there is at least one Adoption Eligibility row WHERE ADOPTION_ELIG.dt_eff_elig (or ADOPTION_ELIG_REDET.dt_eff_elig) <= PAYMENT.dt_pmnt_begin, then continue with **Logic A**.

**Logic A (Adoptions)**

3.1.1.   Select the Adoption Eligibility record WHERE ADOPTION_ELIG.dt_eff_elig (or ADOPTION_ELIG_REDET.dt_eff_elig)<= PAYMENT.dt_pmnt_begin AND ADOPTION_ELIG.dt_cmplt (or ADOPTION_ELIG_REDET.dt_cmplt) is most recent. Store dt_eff_elig and cd_elig_stat at a temporary table.

3.1.1.1.　　Continue to execute the following routine WHILE
ADOPTION_ELIG.dt_eff_elig (or ADOPTION_ELIG_REDET.dt_eff_elig)
of the processed Adoption Eligibility record <= PAYMENT.dt_pmnt_end.
Once all Adoption Eligibility rows are processed for this payment, continue
with **Logic C**.

3.1.1.1.1.　　Select the next Adoption Eligibility record and store
ADOPTION_ELIG.dt_eff_elig (or
ADOPTION_ELIG_REDET.dt_eff_elig) and
ADOPTION_ELIG.cd_elig_stat
(ADOPTION_ELIG_REDET.cd_elig_stat) in the same temporary table
ONLY if satisfies all the criteria listed below. If it fails at least one
criterion, then do not store the Adoption Eligibility record information
and continue to process the next Adoption Eligibility record.

3.1.1.1.1.1.　　ADOPTION_ELIG.dt_eff_elig (or
ADOPTION_ELIG_REDET.dt_eff_elig) of the most recently stored
Eligibility record < ADOPTION_ELIG.dt_eff_elig (or
ADOPTION_ELIG_REDET.dt_eff_elig) of the currently processed
Adoption Eligibility record <= PAYMENT.dt_pmnt_end **AND**

3.1.1.1.1.2.　　ADOPTION_ELIG.dt_eff_elig (or
ADOPTION_ELIG_REDET.dt_eff_elig) is the least recent of the
3.1.1.1.1 criteria **AND**

3.1.1.1.1.3.　　ADOPTION_ELIG.dt_cmplt (or
ADOPTION_ELIG_REDET.dt_cmplt) of the currently processed
Eligibility record > ADOPTION_ELIG.dt_cmplt (or
ADOPTION_ELIG_REDET.dt_cmplt) of the most recently stored
Adoption Eligibility record.

3.2. If there is NO Adoption Eligibility row WHERE ADOPTION_ELIG.dt_eff_elig (or
ADOPTION_ELIG_REDET.dt_eff_elig) <= PAYMENT.dt_pmnt_begin, but there is at
least one Eligibility row WHERE ADOPTION_ELIG.dt_eff_elig (or
ADOPTION_ELIG_REDET.dt_eff_elig) <= PAYMENT.dt_pmnt_end, then continue
with **Logic B**.

## Logic B (Adoptions)

3.2.1. Since there are NO Adoption eligibility records that have
ADOPTION_ELIG.dt_eff_elig (or ADOPTION_ELIG_REDET.dt_eff_elig) <=
PAYMENT.dt_pmnt_begin, the portion of the payment that falls from
PAYMENT.dt_pmnt_begin to ADOPTION_ELIG.dt_eff_elig (or
ADOPTION_ELIG_REDET.dt_eff_elig) of the first available Adoption Eligibility
record will be assumed as being "Pending." Therefore, store
PAYMENT.dt_pmnt_begin and cd_elig_stat = "3" (i.e., Pending) at a temporary

table.

3.2.2. Select the Adoption Eligibility record WHERE ADOPTION_ELIG.dt_eff_elig (or ADOPTION_ELIG_REDET.dt_eff_elig) is least recent and ADOPTION_ELIG.dt_eff_elig (or ADOPTION_ELIG_REDET.dt_eff_elig)<= PAYMENT.dt_pmnt_end. If there are more than one ADOPTION_ELIG or ADOPTION_ELIG_REDET records with the same dt_eff_elig, then select the record with the most recent ADOPTION_ELIG.dt_cmplt (or ADOPTION_ELIG_REDET.dt_cmplt).

3.2.2.1. Continue to execute the following routine WHILE ADOPTION_ELIG.dt_eff_elig (or ADOPTION_ELIG_REDET.dt_eff_elig) (of the processed Adoption Eligibility record) <= PAYMENT.dt_pmnt_end. Once all Adoption Eligibility rows are processed for this payment, continue with **Logic C**.

3.2.2.1.1. Select the next Adoption Eligibility record and store ADOPTION_ELIG.dt_eff_elig (ADOPTION_ELIG_REDET.dt_eff_elig) and ADOPTION_ELIG.cd_elig_stat (or ADOPTION_ELIG_REDET.cd_elig_stat) in the same temporary table ONLY if is satisfies all the criteria listed below. If it fails at least one criterion, then do not store the Adoption Eligibility record information and continue to process the next Adoption Eligibility record.

3.2.2.1.1.1. ADOPTION_ELIG.dt_eff_elig (or ADOPTION_ELIG_REDET.dt_eff_elig) of the most recently stored Adoption Eligibility record < ADOPTION_ELIG.dt_eff_elig (or ADOPTION_ELIG_REDET.dt_eff_elig)of the currently processed Adoption Eligibility record <= PAYMENT.dt_pmnt_end **AND**

3.2.2.1.1.2. ADOPTION_ELIG.dt_eff_elig (or ADOPTION_ELIG_REDET.dt_eff_elig) is least recent of the 3.2.2.1.1 criteria **AND**

3.2.2.1.1.3. ADOPTION_ELIG.dt_cmplt (or ADOPTION_ELIG_REDET.dt_cmplt) of the currently processed Eligibility record > ADOPTION_ELIG.dt_cmplt (or ADOPTION_ELIG_REDET.dt_cmplt)of the most recently stored Adoption Eligibility record.

## Logic C (Adoptions)

3.3. The purpose of Logic C is to identify if a particular payment is FFP or Non-FFP. It also calculates the case count information in each eligibility group to assist the Department for their reimbursement calculations.

3.4. Once Logic A and Logic B are executed, the result set should be stored in a temporary

table with two columns. First column stores effective dates and the second column stores the federal/state eligibility status of the child as of that date.

3.5. Based on the way Logic A and Logic B are executed, the effective dates should be sorted by effective dates in chronological order. The following example shows a possible result set for a payment with PAYMENT.dt_pmnt_begin = 03/01/2000 and PAYMENT.dt_pmnt_end = 03/31/2000.

| Effective Date | Eligibility Status |
|---|---|
| 03/01/2000 | 1 (i.e., Federal) |
| 03/15/2000 | 2 (i.e., State) |
| 03/20/2000 | 1 (i.e., Federal) |

3.6. If a WiSACWIS needs to know a child's eligibility at a certain point in time, it will need to refer to this result set. The child with the above result set will have the following federal/state eligibility statuses:

| Effective Dates | Eligibility Status |
|---|---|
| dt_pmnt_begin – 03/14/2000 | 1 (i.e., Federal) |
| 03/15/2000 – 03/20/2000 | 2 (i.e., State) |
| 03/20/2000 – dt_pmnt_end and add one day | 1 (i.e., Federal) |

3.7. Using id_elig lookup to ADOPTION_ELIG and ADOPTION_ELIG_REDET and if dt_end <> null and dt_end =< dt_pmnt_end then replace in the temporary table dt_pmnt_end with dt_end and add additional row in the temporary table such that dt_end through dt_pmnt_end will be assumed to be pending:

| Effective Dates | Eligibility Status |
|---|---|
| dt_pmnt_begin – 03/15/2000 | 1 (i.e., Eligible and Reimbursable) |
| 03/15/2000 – 03/20/2000 | 3 (i.e., Eligible but Non-reimbursable) |
| 03/20/2000 – dt_end | 1 (i.e., Eligible and Reimbursable) |

| dt_end – dt_pmnt_end and add one day | 4 (i.e., Pending) |
|---|---|

3.8. If only one row is selected from the ADOPTION_ELIG (or ADOPTION_ELIG_REDET) table then the calculation will be dt_pmnt_end – dt-pmnt_begin + 1

3.9. When calculating the Federal, State, and Pending portions of a payment, the number of days that the child spent with these statuses are calculated. If we use the above example, then the following table depicts the number of days for the child in each status.

| **Status** | **Number of Days** |
|---|---|
| Federal | 26 |
| State | 5 |

3.9.1. For the id_pmnt retrieved above, use id_case, id_prsn, dt_pmnt_begin and dt_pmnt_end to perform lookup to ADOPTION_ELIG.dt_eff_elig (or ADOPTION_ELIG_REDET.dt_eff_elig)>= dt_pmnt_begin AND =< dt_pmnt_end. Also if ADOPTION_ELIG.dt_eff_elig (or ADOPTION_ELIG_REDET.dt_eff_elig) NOT = dt_pmnt_begin then also retrieve the previous ADOPTION_ELIG_REDET (or ADOPTION_ELIG) for this id_prsn. From the rows that meet this criteria store in a temporary table: dt_cmplt, dt_eff_elig and cd_elig_stat. Sort these rows ascending by dt_eff_elig and descending by dt_cmplt. Number each row retrieved 1 through n. The perform loop and compare each row. If $dt\_cmplt_n > dt\_cmplt_{(n+1)}$ then discard row (n + 1). If row (n +1) is discarded, then it is not used in the next compare, i.e. compare If $dt\_cmplt_n > dt\_cmplt_{(n+2)}$. If this compare is not true, then row (n + 2) would be kept and used in next compare. Once all rows have been evaluated, test if any row cd_elig_stat = 1. If at least row meets criteria set PAYMENT.cd_ffp = 2. If no row meet criteria, set PAYMENT.cd_ffp = 3.

3.10. Also, insert a record in the PAYMENT_ELIG_DAYS table and set the fields as described below.

3.10.1. PAYMENT_ELIG_DAYS.id_pmnt = PAYMENT.id_pmnt

3.10.2. PAYMENT_ELIG_DAYS.ts_cr = current system date

3.10.3. PAYMENT_ELIG_DAYS.id_cr = Batch ID.

3.10.4. PAYMENT_ELIG_DAYS.qt_elig_reimb = 0.

3.10.5. PAYMENT_ELIG_DAYS.qt_elig_nonreimb = 0.

3.10.6. PAYMENT_ELIG_DAYS.qt_inelig = 0.

3.10.7. PAYMENT_ELIG_DAYS.qt_elig_ssi = 0.

3.10.8. PAYMENT_ELIG_DAYS.qt_pending = the number of days with cd_elig_stat = "3."

3.10.9. PAYMENT_ELIG_DAYS.qt_federal = the number of days with cd_elig_stat = "1."

3.10.10.　　　PAYMENT_ELIG_DAYS.qt_state = the number of days with cd_elig_stat = "2."

3.11.　　　Whenever a PAYMENT_ELIG_DAYS record is inserted, set PAYMENT_ELIG_DAYS.cd_stat = "A." Set the cd_stat = "I" on all other PAYMENT_ELIG_DAYS records where PAYMENT_ELIG_DAYS.id_pmnt = PAYMENT.id_pmnt.

3.12.　　　Continue from (5).

4. This section of the program processes those payments made for children who are not in adoptive placements. Select all ELIGIBILITY and ELIG_REDET rows WHERE PAYMENT.id_case = ELIGIBILITY.id_case (or ELIG_REDET.id_case) AND PAYMENT.id_prsn = ELIGIBILITY.id_prsn (or ELIG_REDET.id_prsn) AND ELIGIBILITY.fl_cmplt (or ELIG_REDET.fl_cmplt) = "Y" AND ELIGIBILITY.fl_void <> "Y" (or ELIG_REDET.fl_void <> "Y".)

4.1. If there is NO Eligibility row WHERE ELIGIBILITY.dt_eff (or ELIG_REDET.dt_eff) <= PAYMENT.dt_pmnt_end THEN PAYMENT is Non-FFP.

4.1.1.　Insert a record in the PAYMENT_ELIG_DAYS table and set the fields as described below.

4.1.1.1.　PAYMENT_ELIG_DAYS.id_pmnt = PAYMENT.id_pmnt

4.1.1.2.　PAYMENT_ELIG_DAYS.ts_cr = current system date

4.1.1.3.　PAYMENT_ELIG_DAYS.id_cr = Batch ID.

4.1.1.4.　PAYMENT_ELIG_DAYS.qt_elig_reimb = 0.

4.1.1.5.　PAYMENT_ELIG_DAYS.qt_elig_nonreimb = 0.

4.1.1.6.　PAYMENT_ELIG_DAYS.qt_inelig = 0.

4.1.1.7.　PAYMENT_ELIG_DAYS.qt_elig_ssi = 0.

4.1.1.8.　PAYMENT_ELIG_DAYS.qt_pending = (PAYMENT.dt_pmnt_end – PAYMENT.dt_pamnt_begin + 1).

4.1.1.9.　PAYMENT_ELIG_DAYS.qt_federal = 0.

4.1.1.10.　PAYMENT_ELIG_DAYS.qt_state = 0.

4.1.2.　Whenever a PAYMENT_ELIG_DAYS record is inserted, set PAYMENT_ELIG_DAYS.cd_stat = "A." Set the cd_stat = "I" on all other PAYMENT_ELIG_DAYS records where PAYMENT_ELIG_DAYS.id_pmnt = PAYMENT.id_pmnt.

4.1.3.  Continue from (2) processing the next payment.

4.2. If there is at least one Eligibility row WHERE ELIGIBILITY.dt_eff (or ELIG_REDET.dt_eff) <= PAYMENT.dt_pmnt_begin, then continue with **Logic A**.

**Logic A (Foster Care)**

5.  Select the Eligibility record WHERE ELIGIBILITY.dt_eff (or ELIG_REDET.dt_eff)<= PAYMENT.dt_pmnt_begin AND ELIGIBILITY.dt_cmplt (or ELIG_REDET.dt_cmplt) is most recent. Store dt_eff, cd_stat and cd_ssi in a temporary table.

5.1.1.  Continue to execute the following routine WHILE ELIGIBILITY.dt_eff (or ELIG_REDET.dt_eff) of the processed Eligibility record <= PAYMENT.dt_pmnt_end. Once all Eligibility rows are processed for this payment, continue with **Logic C**.

5.1.1.1.1.  Select the next Eligibility record and store ELIGIBILITY.dt_eff (or ELIG_REDET.dt_eff) and ELIGIBILITY.cd_stat (ELIG_REDET.cd_stat) in the same temporary table ONLY if satisfies all the criteria listed below. If it fails at least one criterion, then do not store the Eligibility record information and continue to process the next Eligibility record.

5.1.1.1.1.1.  ELIGIBILITY.dt_eff (or ELIG_REDET.dt_eff) of the most recently stored Eligibility record < ELIGIBILITY.dt_eff (or ELIG_REDET.dt_eff) of the currently processed Eligibility record <= PAYMENT.dt_pmnt_end **AND**

5.1.1.1.1.2.  ELIGIBILITY.dt_eff (or ELIG_REDET.dt_eff) is least recent of the 4.2.1.1.1 criteria **AND**

5.1.1.1.1.3.  ELIGIBILITY.dt_cmplt (or ELIG_REDET.dt_cmplt) of the currently processed Eligibility record > ELIGIBILITY.dt_cmplt (or ELIG_REDET.dt_cmplt) of the most recently stored Eligibility record.

5.2. If there is NO Eligibility row WHERE ELIGIBILITY.dt_eff (or ELIG_REDET.dt_eff) <= PAYMENT.dt_pmnt_begin, but there is at least one Eligibility row WHERE ELIGIBILITY.dt_eff (or ELIG_REDET.dt_eff) <= PAYMENT.dt_pmnt_end, then continue with **Logic B**.

**Logic B (Foster Care)**

5.2.1.  Since there are NO eligibility records that have ELIGIBILITY.dt_eff (or ELIG_REDET.dt_eff) <= PAYMENT.dt_pmnt_begin, the portion of the payment that falls from PAYMENT.dt_pmnt_begin to ELIGIBILITY.dt_eff (or ELIG_REDET.dt_eff) of the first available Eligibility record will be assumed as

being "Pending." Therefore, store PAYMENT.dt_pmnt_begin and cd_stat = "4" (i.e., Pending) at a temporary table.

5.2.2. Select the Eligibility record WHERE ELIGIBILITY.dt_eff (or ELIG_REDET.dt_eff) is least recent and ELIGIBILITY.dt_eff (or ELIG_REDET.dt_eff)<= PAYMENT.dt_pmnt_end. If there are more than one ELIGIBILITY or ELIG_REDET records with the same dt_eff, then select the record with the most recent ELIGIBILITY.dt_cmplt (or ELIG_REDET.dt_cmplt).

5.2.2.1. Continue to execute the following routine WHILE ELIGIBILITY.dt_eff (or ELIG_REDET.dt_eff) (of the processed Eligibility record) <= PAYMENT.dt_pmnt_end. Once all Eligibility rows are processed for this payment, continue with **Logic C**.

5.2.2.1.1. Select the next Eligibility record and store ELIGIBILITY.dt_eff (ELIG_REDET.dt_eff) and ELIGIBILITY.cd_stat (or ELIG_REDET.cd_stat) and Eligibility.cd_ssi (or Elig_Redet.cd_ssi) in the same temporary table ONLY if is satisfies all the criteria listed below. If it fails at least one criterion, then do not store the Eligibility record information and continue to process the next Eligibility record.

5.2.2.1.2. ELIGIBILITY.dt_eff (or ELIG_REDET.dt_eff) of the most recently stored Eligibility record < ELIGIBILITY.dt_eff (or ELIG_REDET.dt_eff)of the currently processed Eligibility record <= PAYMENT.dt_pmnt_end **AND**

5.2.2.1.3. ELIGIBILITY.dt_eff (or ELIG_REDET.dt_eff) is least recent of the 4.2.3.1 criteria **AND**

5.2.2.1.4. ELIGIBILITY.dt_cmplt (or ELIG_REDET.dt_cmplt)of the currently processed Eligibility record > ELIGIBILITY.dt_cmplt (or ELIG_REDET.dt_cmplt)of the most recently stored Eligibility record.

**Logic C (Foster Care)**

5.3. The purpose of Logic C is to identify if a particular payment is FFP or Non-FFP. It also calculates the case count information in each eligibility group to assist the Department for their reimbursement calculations.

5.4. Once Logic A and Logic B are executed, the result set should be stored in a temporary table with three columns. The first column stores the effective dates, the second column stores the 'eligibility and reimbursability' status of the child as of that date, and the third column stores the SSI status (for administrative claiming purposes) as of that date.

5.5. Based on the way Logic A and Logic B are executed, the effective dates should be sorted by effective dates in chronological order. The following example shows a possible result set for a payment with PAYMENT.dt_pmnt_begin = 03/01/2000 and PAYMENT.dt_pmnt_end = 03/31/2000.

| Effective Date | Eligibility Status | SSI Status |
|---|---|---|
| 03/01/2000 | 1 (i.e., Eligible and Reimbursable) | N |
| 03/15/2000 | 3 (i.e., Eligible but Non-reimbursable) | Y |
| 03/20/2000 | 1 (i.e., Eligible and Reimbursable) | N |
| 03/26/2000 | 3 (i.e., Eligible but Non-reimbursable) | N |

5.6. If a WiSACWIS needs to know a child's eligibility at a certain point in time, it will need to refer to this result set. The child with the above result set will have the following Eligibility and Reimbursability statuses:

| Effective Dates | Eligibility Status |
|---|---|
| dt_pmnt_begin – 03/15/2000 | 1 (i.e., Eligible and Reimbursable) |
| 03/15/2000 – 03/20/2000 | 5 (i.e., Eligible and Receiving SSI) - Note that the status has changed from 3 to 5 because of SSI. |
| 03/20/2000 – 3/26/2000 | 1 (i.e., Eligible and Reimbursable) |
| 03/26/2000 – dt_pmnt_end and add one day | 3 (i.e., Eligible but Non-reimbursable. Not receiving SSI) |

5.7. Using id_elig lookup to ELIGIBILITY and ELIG_REDET and if dt_end <> null and dt_end =< dt_pmnt_end then replace in the temporary table dt_pmnt_end with dt_end and add additional row in the temporary table such that dt_end through dt_pmnt_end will be assumed to be pending:

| Effective Dates | Eligibility Status |
|---|---|
| dt_pmnt_begin – 03/15/2000 | 1 (i.e., Eligible and Reimbursable) |
| 03/15/2000 – 03/20/2000 | 5 (i.e., Eligible and Receiving SSI) - Note that the status has changed from 3 to 5 because of SSI. |

| 03/20/2000 – 3/26/2000 | 1 (i.e., Eligible and Reimbursable) |
|---|---|
| 03/26/2000 – dt_end | 3 (i.e., Eligible but Non-reimbursable. Not receiving SSI) |
| Dt_end - dt_pmnt_end and add one day | 4 (i.e., Pending) |

5.8. If only one row is selected from the ELIGIBILITY (or ELIG_REDET) table then the calculation will be dt_pmnt_end – dt-pmnt_begin + 1

5.9. When calculating the "Eligible and Reimbursable", "Eligible but Non-reimbursable", "Ineligible", 'Eligible and Receiving SSI' and "Pending" portions of a payment, the number of days that the child spent with these statuses are calculated. If we use the above example, then the following table depicts the number of days for the child in each status.

| Status | Number of Days |
|---|---|
| Eligible and Reimbursable | 20 |
| Eligible but Non-reimbursable (Not Receiving SSI). | 5 |
| Eligible but Receiving SSI | 6 |

5.10.    For the id_pmnt retrieved above, use id_case, id_prsn, dt_pmnt_begin and dt_pmnt_end perform lookup to ELIGIBLITY.dt_eff_elig (or ELIG_REDET.dt_eff_elig)>= dt_pmnt_begin AND =< dt_pmnt_end. Also if ELIGIBILITY.dt_eff_elig (or ELIG_REDET.dt_eff_elig) NOT = dt_pmnt_begin then also retrieve the previous ELIG_REDET (or ELIGIBILITY) for this id_prsn.  From the rows that meet this criteria store in a temporary table:  dt_cmplt, dt_elig and cd_stat. Sort these rows ascending by dt_eff_elig and descending by dt_cmplt.  Number each row retrieved 1 through n.  The perform loop and compare each row.  If $dt\_cmplt_n >$ $dt\_cmplt_{(n + 1)}$ then discard row (n + 1).  If row (n +1) is discarded, then it is not used in the next compare, i.e. compare If $dt\_cmplt_n > dt\_cmplt_{(n + 2)}$ .  If this compare is true, then row (n + 2) will be kept and used in next compare. Once all rows have been evaluated, test if any row cd_stat = 1.  If at least row meets criteria set PAYMENT.cd_ffp = 2.  If no row meet criteria, set PAYMENT.cd_ffp = 3.

5.11.    Also, insert a record in the PAYMENT_ELIG_DAYS table and set the fields as described below.

5.11.1. PAYMENT_ELIG_DAYS.id_pmnt = PAYMENT.id_pmnt

5.11.2. PAYMENT_ELIG_DAYS.ts_cr = current system date

  5.11.3. PAYMENT_ELIG_DAYS.id_cr = Batch ID.

  5.11.4. PAYMENT_ELIG_DAYS.qt_elig_reimb = the number of days with cd_stat = "1."

  5.11.5. PAYMENT_ELIG_DAYS.qt_elig_nonreimb = the number of days with cd_stat = "2."

  5.11.6. PAYMENT_ELIG_DAYS.qt_inelig = the number of days with cd_stat = "3."

  5.11.7. PAYMENT_ELIG_DAYS.qt_elig_ssi = the number of days with cd_stat = "5."

  5.11.8. PAYMENT_ELIG_DAYS.qt_pending = = the number of days with cd_stat = "4."

  5.11.9. PAYMENT_ELIG_DAYS.qt_federal = 0.

  5.11.10.  PAYMENT_ELIG_DAYS.qt_state = 0.

 5.12.  Whenever a PAYMENT_ELIG_DAYS record is inserted, set PAYMENT_ELIG_DAYS.cd_stat = "A." Set the cd_stat = "I" on all other PAYMENT_ELIG_DAYS records where PAYMENT_ELIG_DAYS.id_pmnt = PAYMENT.id_pmnt.

6. After each update to PAYMENT.cd_ffp, SET fl_ffp_calc= "N."

7. If PAYMENT.cd_ffp is updated with "2" where it was previously "3" OR PAYMENT.cd_ffp is updated with "3" where it was previously "3" OR

 PAYMENT.cd_ffp is updated with "3" where it was previously "2" OR PAYMENT.cd_ffp is updated with "2", where it was previously "2", then UPDATE PAYMENT.fl_ive_calc = "Y."

Stop when all PAYMENT record selected in (1) are processed.

### 1.4.3. <u>Title IV-E Reimbursement Batch Program</u>

**Program Name:**    b-fm04a-reimbursement

**Process Summary:**    This batch program determines the portion of each payment that needs to be claimed using Title IV-E funds.  Payments are categorized as one of three types:

1)  Title IV-E Regular,

2)  Non-Title IV-E,

3)  Title IV-E (50/50).

Using the non-trust amount from a payment, this batch will first determine if a payment is in fact eligible for federal reimbursement and that the service type is either (1) or (3).  If at least one day of the payment falls within a period that is earmarked as Eligible and Reimbursable or Federal (for adoptions in-home services), it will apply federal percentages.  The percentages applied are prorated based on the number of days a payment is federally reimbursable within a payment's begin date and end date.  For payments to Child Caring Institution (CCI) or Group Home providers, the batch will apply the Maintenance Percentage (recorded in the Provider's Eligibility Cost page) and then apply the FFP percentage (recorded in the Maintain Title IV-E FFP page) for those payments that have a service type of (1).  For all other provider, the batch will only apply the FFP percentage for those payment that have a service type of (1). If a payment has a service type of (3), then a 50/50 split is applied instead of the FFP percentage.  If the provider is a CCI, the batch will further calculate what portion of the payment is eligible for Title XIX reimbursement.   Please read the below Process Description section for the details of these calculations.

Payments with the statewide reporting group code of '999' will not be processed by the FFP batch.  These payments are non-system disbursed, one-time payments issued for pre-SACWIS cost of care.

**Frequency:**    Monthly.

**Dependencies:**    Successful completion of Trust Account Draw-Down batch program batch program.

**Input Parameters:**   From Date.


**Input Files:**   cycle-date-override


**Output Files:**   None


**Database Tables:**
| | | |
|---|---|---|
| PAYMENT | | RU |
| ELIGIBILITY | | RU |
| ELIG_REDET | RU | |
| SERVICE_TYPE | | R |
| PAYMENT_REIMBURSEMENT | | CU |
| IV-FFP | | R |
| PRVD_ELIG_COST | | R |


**Process Description:**


This batch program processes those payments that are potentially reimbursable by Title IV-E and calculates the amounts that need to be claimed. Prior to this program's run, the trust funds have already been deducted from the payments and the payments have already been identified as being FFP or Non-FFP. Based on this information, this batch program will calculate the dollar amount (if any) that needs to be claimed from Title IV-E funds for each payment.  Finally, this process also creates the necessary information to generate the reports that will assist the department in its Title IV-E claming.


In detail, the Federal Amount located in the Title IV-E Reimbursement History box is calculated using one of three methods. The first two equations are applied only if the payment has at least one day of a status of Eligible and Reimbursable or Federal as indicated in the Days of Care History box.  The first method is applied for out-of-home care eligibility payments that have a service type of Title IV-E (Regular). The second method is applied for adoption assistance eligibility payments that also have a service type of Title IV-E (Regular).  The third method is applied for adoption assistance eligibility payments that have a service type of Title IV-E (50/50).  This federal assistance is applied regardless of child's eligibility.

1) <u>IV-E Federal Amount</u>

$$IV\text{-}E\ Federal\ Amount = Non\text{-}Trust\ Amount \times$$
$$IV\text{-}E\ Prorated\ Percentage \times$$
$$Maintenance\ Percentage \times$$
$$FFP\ Percentage$$

where,

$$\text{IV - E Prorated Percentage} = \frac{\text{Elig Reimb Days}}{\text{Elig Reimb Days} + \text{Elig Nonreimb Days} + \text{Ineligible Days} + \text{Pending Days}}$$

2) <u>Adoption Assistance Federal Amount</u>

$$\text{Adoption Federal Amount} = \text{Non - Trust Amount} \times \\ \text{Adoption Prorated Percentage} \times \\ \text{FFP Percentage}$$

where,

$$\text{Adopton Prorated Percentage} = \frac{\text{Federal Days}}{\text{Federal Days} + \text{State Days} + \text{Pending Days}}$$

3) <u>Adoption Assistance 50/50 Federal Amount</u>

$$\text{Adoption Federal Amount} = \text{Non - Trust Amount} \times \\ \text{50 Percent}$$

If payment does not have at least at least one day of Eligible and Reimbursable or Federal, as displayed in the Days of Care History box on the Payment Reimbursement History page, then the State is obligated to pay the entire non-trust amount.

The State Amount located in the Title IV-E Reimbursement History box is derived from the following equation.

- State Amount = Non-Trust Amount - Federal Amount

The Title XIX amount is derived from the following equation and applied only if a payment was for a CCI:

- Title XIX Amount = Non-Trust Amount X Title XIX percentage X FFP Percentage

The Maintenance Percentage, State Percentage and Title XIX Percentage mentioned above are based on the Provider's Eligibility Cost page (*PM02b: Private Provider Case File*). These percentages are only applied to Private Providers, and the following equation must be maintained.

- 100% (Not Displayed) = Maintenance Percentage + State Percentage (Percentage (Displayed on Provider's Eligibility Cost page) + Title XIX Percentage

Home Providers only have the FFP percentage applied to any Non-Trust Amount, if the correct eligibility criteria are met.

If a payment does not have at least one day of a status of Eligible and Reimbursable or Federal as indicated in the Days of Care History box, then the State is responsible to pay any Non-Trust Amount.

**Processing:**

<u>**Section A**</u>

This section of the program calculates the IV-E reimbursable portion of each payment made for a placement. It also re-calculates the Title IV-E reimbursable portion of those payments that had a retroactive change in their FFP status.

1.  Select system disbursed (+) payments. Select all PAYMENT records where PAYMENT.am_rqst > 0 AND PAYMENT.cd_pmnt_stat <> "C" AND PAYMENT.dt_chck >= From Date input parameter and SERVICE_TYPE.cd_ive_12 <>"3" (i.e., Title IV-E (50/50)) and SERVICE_TYPE.cd_ive_12 <> 4 (code value for 'exclude from claiming') for the selected PAYMENT.cd_srvc and PAYMENT.fl_ive_calc = "Y" AND PAYMENT.cd_plcm_setng is NOT NULL AND SERVICE_TYPE.fl_non_sys_disb = "N" for the PAYMENT.cd_srvc selected

2.  Select non-system disbursed (+) payments. Select all **approved** PAYMENT records where PAYMENT.am_rqst > 0 AND PAYMENT.cd_pmnt_stat <> "C" AND PAYMENT.dt_pmnt_end >= From Date input parameter and SERVICE_TYPE.cd_ive_12 <> "3" (i.e., Title IV-E (50/50)) for the selected PAYMENT.cd_srvc and PAYMENT.fl_ive_calc = "Y" AND PAYMENT.cd_plcm_setng is NOT NULL AND SERVICE_TYPE.fl_non_sys_disb = "Y" for the PAYMENT.cd_srvc selected.

3.  Select (-) payments. Select all **approved** PAYMENT records WHERE PAYMENT.cd_pmnt_stat <> "C" AND PAYMENT.cd_plcm_setng is NOT NULL AND PAYMENT.am_rqst < 0 AND SERVICE_TYPE.cd_ive_12 <> "3" (i.e., Title IV-E (50/50)) AND SERVICE_TYPE.cd_ive_12 <> 4 (code value for 'exclude from claiming') for the selected PAYMENT.cd_ srvc and PAYMENT.fl_ive_calc = "Y" AND there is **no** TRUST_ACCOUNT_LEDGER record WHERE TRUST_ACCOUNT_LEDGER.id_prsn = PAYMENT.id_prsn AND TRUST_ACCOUNT_LEDGER.cd_detl = the code value for "Reimbursement" and TRUST_ACCOUNT_LEDGER.am_trans < 0 AND  perform lookup with the selected TRUST_ACCOUNT_LEDGER.id_pmnt to PAYMENT.id_pmnt and verify that no row has the same month as the month of the payment (-).  Do not select payments where SERVICE_TYPE.cd_sw_rpt_grp = '999'.

4.  Select the first PAYMENT record and read PAYMENT.cd_ffp.

    4.1. If PAYMENT.cd_ffp = "2" (i.e., FFP), then insert two records into PAYMENT_REIMBURSEMENT table and populate the fields with data as described below.  A third record will sometimes be inserted if CODE_DESC.cd_cdesc_type = "2" (i.e., CCI) = (WHERE CODE_DESC.id_grp = "PLCMSTNG" AND CODE_DESC.id_grpi = PAYMENT.cd_plcm_setng.)

        4.1.1. **For the first record**, Set PAYMENT_REIMBURSEMENT.id_pmnt = PAYMENT.id_pmnt

        4.1.2. PAYMENT_REIMBURSEMENT.ts_cr = Current system date

        4.1.3. PAYMENT_REIMBURSEMENT.id_cr = Batch ID

4.1.4.  PAYMENT_REIMBURSEMENT.cd_reimb_type = "1" (i.e., Federal) OR "4" (i.e., TANF) if cd_cdesc_type = "4" (i.e., Kinship)

4.1.5.  PAYMENT_REIMBURSEMENT.am_reimb =

4.1.5.1.   If CODE_DESC.cd_cdesc_type <> "2" (i.e., CCI) OR "1" (i.e., Group Home) OR "4" (i.e., Kinship) (WHERE CODE_DESC.id_grp = "PLCMSTNG" AND CODE_DESC.id_grpi = PAYMENT.cd_plcm_setng.

- Then perform lookup with id_pmnt to PAYMENT_ELIG_DAYS and if qt-elig_reimb > 0, then divide qt_elig_reimb by (qt_elig_reimb + qt_elig_nonreimb + qt_inelig + qt_pending).  Store result in WS-PRCT-PYMT.   Otherwise, if qt_federal > 0, then divide qt_federal by (qt_federal + qt_state + qt_pending).   Store result in WS-PRCT-PYMT.  (ie only qt_federal or qt_elig_reimb will have a value greater than zero)

- Then Multiply WS-PRCT-PYMT by PAYMENT.am_nontrust (multiply with PAYMENT.am_rqst – PAYMENT.am_admin for (-) payments), and store value in WS-FED-AMT.

- Then access the IVE_FFP table and Select IVE_FFP.am_ffp WHERE IVE_FFP.ts_cr is most recent and IVE_FFP.dt_eff <= PAYMENT.dt_pmnt_begin (this will pick up the most recently entered IV-E FFP percentage that was effective at the time of service begin date). Multiply IVE_FFP.am_ffp with WS-FED-AMT.

- Update PAYMENT_REIMBURSEMENT.am_reimb with the result.

4.1.5.2.   If CODE_DESC.cd_cdesc_type = "2" (i.e., CCI) OR "1" (i.e., Group Home) WHERE CODE_DESC.id_grp = "PLCMSTNG" AND CODE_DESC.id_grpi = PAYMENT.cd_plcm_setng,

- Then perform lookup with id_pmnt to PAYMENT_ELIG_DAYS where cd_stat = 'A' and if qt-elig_reimb > 0, then divide qt_elig_reimb by (qt_elig_reimb + qt_elig_nonreimb + qt_inelig + qt_pending).  Store result in WS-PRCT-PYMT.   Otherwise, if qt_federal > 0, then divide qt_federal by (qt_federal + qt_state + qt_pending).   Store result in WS-PRCT-PYMT. (ie only qt_federal or qt_elig_reimb will have a value greater than zero)

- Then Multiply WS-PRCT-PYMT by PAYMENT.am_nontrust (multiply with PAYMENT.am_rqst – PAYMENT.am_admin for (-) payments), and store value in WS-FED-AMT.

- Then Access the PRVD_ELIG_COST table using PAYMENT.cd_srvc and PAYMENT.id_prvd_org. Select PRVD_ELIG_COST.am_maint WHERE PRVD_ELIG_COST.ts_cr is most recent and PRVD_ELIG_COST.dt_eff <= PAYMENT.dt_pmnt_begin (this will pick up the most recently entered percentage that was effective at the time of service begin date). If there is no row in the PRVD_ELIG_COST table to be selected then use PRVD_ELIG_COST.am_maint as "1." Multiply WS-FED-AMT with PRVD_ELIG_COST.am_maint. Store this amount as WS-maint-Amount.

- Then Access the IVE_FFP table and Select IVE_FFP.am_ffp WHERE
  IVE_FFP.ts_cr is most recent and IVE_FFP.dt_eff <=
  PAYMENT.dt_pmnt_begin (this will pick up the most recently entered IV-
  E FFP percentage that was effective at the time of service begin date).
  Multiply IVE_FFP.am_ffp with the WS-maint-Amount.

- Update PAYMENT_REIMBURSEMENT.am_reimb with the result.

    4.1.5.3.    If CODE_DESC.cd_cdesc_type = "4" (i.e., Kinship) (WHERE
CODE_DESC.id_grp = "PLCMSTNG" AND CODE_DESC.id_grpi =
PAYMENT.cd_plcm_setng.

- PAYMENT.am_nontrust (or PAYMENT.am_rqst –
  PAYMENT.am_admin for (-) payments),

- Update PAYMENT_REIMBURSEMENT.am_reimb with the result.

4.1.6.  PAYMENT_REIMBURSEMENT.am_titlexix = NULL

4.1.7.  PAYMENT_REIMBURSEMENT.am_ffp_pct= Refer to
PAYMENT_REIMBURSEMENT.am_reimb field for the details of the retrieval.

4.1.8.  PAYMENT_REIMBURSEMENT.am_titlexix_pct = NULL.

4.1.9.  PAYMENT_REIMBURSEMENT.am_maint_pct = Refer to
PAYMENT_REIMBURSEMENT.am_reimb field for the details of the retrieval.

4.1.10. **For the second record,** Set PAYMENT_REIMBURSEMENT.id_pmnt =
PAYMENT.id_pmnt

4.1.11. PAYMENT_REIMBURSEMENT.ts_cr = Current system date

4.1.12. PAYMENT_REIMBURSEMENT.id_cr = Batch ID

4.1.13. PAYMENT_REIMBURSEMENT.cd_reimb_type = "2" (i.e., State).

4.1.14. PAYMENT_REIMBURSEMENT.am_reimb = (PAYMENT.am_nontrust –
PAYMENT_REIMBURSEMENT.am_reimb of the first record inserted) – WS-
TITLEXIX-AMT (see calculation below). Set this field to [(PAYMENT.am_rqst –
am_admin) – PAYMENT_REIMBURSEMENT.am_reimb of the first record
inserted] for (-) payments – WS-TITLEXIX-AMT (see calculation below)

4.1.15. PAYMENT_REIMBURSEMENT.am_titlexix = NULL

4.1.16. PAYMENT_REIMBURSEMENT.am_ffp_pct= NULL.

4.1.17. PAYMENT_REIMBURSEMENT.am_titlexix_pct = NULL

4.1.18. PAYMENT_REIMBURSEMENT.am_maint_pct = NULL.

4.1.19. **For the third record,** Set PAYMENT_REIMBURSEMENT.id_pmnt =
PAYMENT.id_pmnt

4.1.20. PAYMENT_REIMBURSEMENT.ts_cr = Current system date

4.1.21. PAYMENT_REIMBURSEMENT.id_cr = Batch ID

4.1.22. PAYMENT_REIMBURSEMENT.cd_reimb_type = "3" (i.e., Title XIX).

4.1.23. PAYMENT_REIMBURSEMENT.am_reimb = null

4.1.24. PAYMENT_REIMBURSEMENT.am_titlexix =

    4.1.24.1.   If CODE_DESC.cd_cdesc_type <> "2" (i.e., CCI) WHERE CODE_DESC.id_grp = "PLCMSTNG" AND CODE_DESC.id_grpi = PAYMENT.cd_plcm_setng, then set to NULL. However, if CODE_DESC.cd_cdesc_type = "2" (i.e., CCI) WHERE  CODE_DESC.id_grp = "PLCMSTNG" AND CODE_DESC.id_grpi = PAYMENT.cd_plcm_setng, then access the PRVD_ELIG_COST table using PAYMENT.cd_srvc and PAYMENT.id_payee. Select PRVD_ELIG_COST.am_titlexix WHERE PRVD_ELIG_COST.ts_cr is most recent and PRVD_ELIG_COST.dt_eff <= PAYMENT.dt_pmnt_begin (this will pick up the most recently entered percentage that was effective at the time of service begin date). If there is no row in the PRVD_ELIG_COST table to be selected then use PRVD_ELIG_COST.am_titlexix as "0." Update PAYMENT_REIMBURSEMENT.am_titlexix_pct with the selected PRVD_ELIG_COST.am_titlexix. Multiply PAYMENT.am_nontrust (multiply with PAYMENT.am_rqst – PAYMENT.am_admin for (-) payments) with PRVD_ELIG_COST.am_titlexix_pct and multiply by IVE_FFP.am_ffp. Update PAYMENT_REIMBURSEMENT.am_titlexix and WS-TITLEXIX-AMT with the result

4.1.25. PAYMENT_REIMBURSEMENT.am_ffp_pct = IVE_FFP.am_ffp.  Refer to PAYMENT_REIMBURSEMENT.am_reimb field for the details of the retrieval..

4.1.26. PAYMENT_REIMBURSEMENT.am_titlexix_pct = PRVD_ELIG_COST.am_titlexix.  Refer to PAYMENT_REIMBURSEMENT.am_titlexix field for the details of the retrieval.

4.1.27. PAYMENT_REIMBURSEMENT.am_maint_pct = NULL.

4.2. Set PAYMENT.fl_ive_calc = "N."

4.3. If PAYMENT.cd_ffp = "3" (i.e., Non-FFP), then insert one record into PAYMENT_REIMBURSEMENT table and populate its fields with data as described below. A second record will sometimes be inserted if CODE_DESC.cd_cdesc_type = "2" (i.e., CCI) = (WHERE CODE_DESC.id_grp = "PLCMSTNG" AND CODE_DESC.id_grpi = PAYMENT.cd_plcm_setng.)

    4.3.1.   PAYMENT_REIMBURSEMENT.id_pmnt = PAYMENT.id_pmnt

    4.3.2.   PAYMENT_REIMBURSEMENT.ts_cr = Current system date

    4.3.3.   PAYMENT_REIMBURSEMENT.id_cr = Batch ID

    4.3.4.   PAYMENT_REIMBURSEMENT.cd_reimb_type = "2" (i.e., State) OR "4" (i.e., TANF) if cd_cdesc_type = "4" (i.e., Kinship)

    4.3.5.   PAYMENT_REIMBURSEMENT.am_reimb = PAYMENT.am_nontrust (PAYMENT.am_rqst – PAYMENT.am_admin for (-) payments) – WS-TITLEXIX-

AMT (see calculation below)

4.3.6. PAYMENT_REIMBURSEMENT.am_titlexix = NULL

4.3.7. PAYMENT_REIMBURSEMENT.am_ffp_pct= NULL.

4.3.8. PAYMENT_REIMBURSEMENT.am_titlexix_pct = NULL

4.3.9. PAYMENT_REIMBURSEMENT.am_maint_pct = NULL.

4.3.10. **For the second record,** Set PAYMENT_REIMBURSEMENT.id_pmnt = PAYMENT.id_pmnt

4.3.11. PAYMENT_REIMBURSEMENT.ts_cr = Current system date

4.3.12. PAYMENT_REIMBURSEMENT.id_cr = Batch ID

4.3.13. PAYMENT_REIMBURSEMENT.cd_reimb_type = "3" (i.e., Title XIX).

4.3.14. PAYMENT_REIMBURSEMENT.am_reimb = null

4.3.15. PAYMENT_REIMBURSEMENT.am_titlexix =

4.3.15.1.   If CODE_DESC.cd_cdesc_type <> "2" (i.e., CCI) WHERE CODE_DESC.id_grp = "PLCMSTNG" AND CODE_DESC.id_grpi = PAYMENT.cd_plcm_setng, then set to NULL. However, if CODE_DESC.cd_cdesc_type = "2" (i.e., CCI) WHERE  CODE_DESC.id_grp = "PLCMSTNG" AND CODE_DESC.id_grpi = PAYMENT.cd_plcm_setng, then access the PRVD_ELIG_COST table using PAYMENT.cd_srvc and PAYMENT.id_payee. Select PRVD_ELIG_COST.am_titlexix WHERE PRVD_ELIG_COST.ts_cr is most recent and PRVD_ELIG_COST.dt_eff <= PAYMENT.dt_pmnt_begin (this will pick up the most recently entered percentage that was effective at the time of service begin date). If there is no row in the PRVD_ELIG_COST table to be selected then use PRVD_ELIG_COST.am_titlexix as "0." Update PAYMENT_REIMBURSEMENT.am_titlexix_pct with the selected PRVD_ELIG_COST.am_titlexix. Multiply PAYMENT.am_nontrust (multiply with PAYMENT.am_rqst – PAYMENT.am_admin for (-) payments) with PRVD_ELIG_COST.am_titlexix_pct and multiply by IVE_FFP.am_ffp. Update PAYMENT_REIMBURSEMENT.am_titlexix with the result.

4.3.16. PAYMENT_REIMBURSEMENT.am_ffp_pct = IVE_FFP.am_ffp.  Refer to PAYMENT_REIMBURSEMENT.am_reimb field for the details of the retrieval..

4.3.17. PAYMENT_REIMBURSEMENT.am_titlexix_pct = PRVD_ELIG_COST.am_titlexix.  Refer to PAYMENT_REIMBURSEMENT.am_titlexix field for the details of the retrieval.

4.3.18. PAYMENT_REIMBURSEMENT.am_maint_pct = NULL.

4.4. Set PAYMENT.fl_ive_calc = "N."

4.5. Whenever a PAYMENT_REIMBURSEMENT record is inserted, set PAYMENT_REIMBURSEMENT.cd_stat = "A." Set the cd_stat = "I" on all other

PAYMENT_REIMBURSEMENT records where
PAYMENT_REIMBURSEMENT.id_pmnt = PAYMENT.id_pmnt.

## Section B

This section of the program calculates the IV-E eligible portion of payments that are paid based on the "50/50" scheme. These are mostly pre-adoption expenses, where 50% of the cost is paid by Title IV-E funds and the rest is paid by State funds.

1. Select system disbursed (+) payments. Select all PAYMENT records WHERE PAYMENT.am_rqst > 0 AND PAYMENT.cd_pmnt_stat <> "C" AND PAYMENT.dt_chck >= From Date input parameter AND SERVICE_TYPE.cd_ive_12 = "3" (i.e., Title IV-E (50/50)) for the selected PAYMENT.cd_srvc AND SERVICE_TYPE.fl_non_sys_disb = "N" for the selected PAYMENT.cd_srvc AND PAYMENT.fl_ive_calc = "Y".

2. Select non-system disbursed (+) payments. Select all **approved** PAYMENT records WHERE PAYMENT.am_rqst > 0 AND PAYMENT.cd_pmnt_stat <> "C" AND PAYMENT.dt_pmnt_end >= From Date input parameter AND SERVICE_TYPE.cd_ive_12 = "3" (i.e., Title IV-E (50/50)) for the selected PAYMENT.cd_srvc AND SERVICE_TYPE.fl_non_sys_disb = "Y" for the selected PAYMENT.cd_srvc AND PAYMENT.fl_ive_calc = "Y".

   2.1. Insert two records in the PAYMENT_REIMBURSEMENT table. Both records will have the same data populated except the PAYMENT.cd_reimb_type. The records are populated as described below.

   2.1.1.  PAYMENT_REIMBURSEMENT.id_pmnt = PAYMENT.id_pmnt

   2.1.2.  PAYMENT_REIMBURSEMENT.ts_cr = Current system date

   2.1.3.  PAYMENT_REIMBURSEMENT.id_cr = Batch ID

   2.1.4.  PAYMENT_REIMBURSEMENT.cd_reimb_type = "1" (i.e., Federal) for the first record and "2" (i.e., State) for the second record.

   2.1.5.  PAYMENT_REIMBURSEMENT.am_reimb = (PAYMENT.am_nontrust / 2)

   2.1.6.  PAYMENT_REIMBURSEMENT.am_titlexix = NULL.

   2.1.7.  PAYMENT_REIMBURSEMENT.am_ffp_pct= 0.50 only for the row with cd_reimb_type = "1" (i.e., Federal)

   2.1.8.  PAYMENT_REIMBURSEMENT.am_titlexix_pct = NULL.

   2.1.9.  PAYMENT_REIMBURSEMENT.am_maint_pct = NULL.

   2.2. Update the selected PAYMENT record with PAYMENT.fl_ive_calc = "N."

   2.3. Whenever a PAYMENT_REIMBURSEMENT record is inserted, set PAYMENT_REIMBURSEMENT.cd_stat = "A." Set the cd_stat = "I" on all other PAYMENT_REIMBURSEMENT records where PAYMENT_REIMBURSEMENT.id_pmnt = PAYMENT.id_pmnt.

2.4. Continue from (1) until all (+) payments are processed.

3. Select (-) payments. Select all **approved** PAYMENT records WHERE PAYMENT.am_rqst < 0 AND SERVICE_TYPE.cd_ive_12 value for the selected PAYMENT.cd_srvc = "3" (i.e., Title IV-E (50/50)) and PAYMENT.fl_ive_calc = "Y" AND there is **no** TRUST_ACCOUNT_LEDGER record WHERE TRUST_ACCOUNT_LEDGER.id_prsn = PAYMENT.id_prsn AND TRUST_ACCOUNT_LEDGER.cd_detl = the code value for "Reimbursement" and TRUST_ACCOUNT_LEDGER.am_trans < 0 AND perform lookup with the selected TRUST_ACCOUNT_LEDGER.id_pmnt to PAYMENT.id_pmnt and verify that no row has the same month as the month of the payment (-).

   3.1. Insert two records in the PAYMENT_REIMBURSEMENT table. Both records will have the same data populated except the PAYMENT.cd_reimb_type. The records are populated as described below.

      3.1.1.  PAYMENT_REIMBURSEMENT.id_pmnt = PAYMENT.id_pmnt

      3.1.2.  PAYMENT_REIMBURSEMENT.ts_cr = Current system date

      3.1.3.  PAYMENT_REIMBURSEMENT.id_cr = Batch ID

      3.1.4.  PAYMENT_REIMBURSEMENT.cd_reimb_type = "1" (i.e., Federal) for the first record and "2" (i.e., State) for the second record.

      3.1.5.  PAYMENT_REIMBURSEMENT.am_reimb = [(PAYMENT.am_rqst – PAYMENT.am_admin) / 2]

      3.1.6.  PAYMENT_REIMBURSEMENT.am_titlexix = NULL.

      3.1.7.  PAYMENT_REIMBURSEMENT.am_ffp_pct= 0.50 only for the row with cd_reimb_type = "1" (i.e., Federal)

      3.1.8.  PAYMENT_REIMBURSEMENT.am_titlexix_pct = NULL.

      3.1.9.  PAYMENT_REIMBURSEMENT.am_maint_pct = NULL.

   3.2. Update the selected PAYMENT record with PAYMENT.fl_ive_calc = "N."

   3.3. Whenever a PAYMENT_REIMBURSEMENT record is inserted, set PAYMENT_REIMBURSEMENT.cd_stat = "A." Set the cd_stat = "I" on all other PAYMENT_REIMBURSEMENT records where PAYMENT_REIMBURSEMENT.id_pmnt = PAYMENT.id_pmnt.

   3.4. Continue from (3) until all (-) payments are processed.


## Section C

This section of the program processes non-placement based payments.

1. Select system disbursed (+) payments. Select all PAYMENT records WHERE PAYMENT.am_rqst > 0 AND PAYMENT.cd_pmnt_stat <> "C" AND PAYMENT.am_non_trust > 0 AND PAYMENT.cd_plcm_setng is NULL AND PAYMENT.dt_chck >= From Date input parameter AND SERVICE_TYPE.cd_ive_12 <>

"3" (i.e., Title IV-E (50/50)) for the selected PAYMENT.cd_srvc AND SERVICE_TYPE.fl_non_sys_disb = "N" for the selected PAYMENT.cd_srvc AND PAYMENT.fl_ive_calc = "Y".

2. Select non-system disbursed (+) payments. Select all **approved** PAYMENT records WHERE PAYMENT.am_rqst > 0 AND PAYMENT.cd_pmnt_stat <> "C" AND PAYMENT.am_non_trust > 0 AND PAYMENT.dt_pmnt_end >= From Date input parameter AND PAYMENT.cd_plcm_setng is NULL AND SERVICE_TYPE.cd_ive_12 <> "3" (i.e., Title IV-E (50/50)) for the selected PAYMENT.cd_srvc AND SERVICE_TYPE.fl_non_sys_disb = "Y" for the selected PAYMENT.cd_srvc AND PAYMENT.fl_ive_calc = "Y".

3. Select (-) payments. Select all **approved** PAYMENT records WHERE PAYMENT.am_rqst < 0 AND SERVICE_TYPE.cd_ive_12 value <> "3" (i.e., Title IV-E (50/50)) for the selected PAYMENT.cd_srvc and PAYMENT.fl_ive_calc = "Y" AND PAYMENT.cd_plcm_setng is NULL AND there is **no** TRUST_ACCOUNT_LEDGER record WHERE TRUST_ACCOUNT_LEDGER.id_prsn = PAYMENT.id_prsn AND TRUST_ACCOUNT_LEDGER.cd_detl = the code value for "Reimbursement" and TRUST_ACCOUNT_LEDGER.am_trans < 0 AND perform lookup with TRUST_ACCOUNT_LEDGER.id_pmnt to PAYMENT.id_pmnt and verify that no row has the same month as the month of the payment (-).

4. Select the first PAYMENT record, IF PAYMENT.cd_ffp = "2" (i.e., Federal), then Insert two records in the PAYMENT_REIMBURSEMENT table. If PAYMENT.cd_ffp = "3", Insert one record in the PAYMENT_REIMBURSEMENT table and continue from (4.5).

   4.1. The **first PAYMENT_REIMBURSEMENT record** is populated as described below.

      4.1.1. PAYMENT_REIMBURSEMENT.id_pmnt = PAYMENT.id_pmnt

      4.1.2. PAYMENT_REIMBURSEMENT.ts_cr = Current system date

      4.1.3. PAYMENT_REIMBURSEMENT.id_cr = Batch ID.

      4.1.4. PAYMENT_REIMBURSEMENT.cd_reimb_type = "1" (i.e., Federal)

      4.1.5. PAYMENT_REIMBURSEMENT.am_reimb =

         - Then perform lookup with id_pmnt to PAYMENT_ELIG_DAYS where cd_stat = 'A' and if qt-elig_reimb > 0, then divide qt_elig_reimb by (qt_elig_reimb + qt_elig_nonreimb + qt_inelig + qt_pending). Store result in WS-PRCT-PYMT. Otherwise, if qt_federal > 0, then divide qt_federal by (qt_federal + qt_state + qt_pending). Store result in WS-PRCT-PYMT. (ie only qt_federal or qt_elig_nonreimb will have a value greater than zero)

         - Then Multiply WS-PRCT-PYMT by PAYMENT.am_nontrust (multiply with PAYMENT.am_rqst – PAYMENT.am_admin for (-) payments), and store value in WS-FED-AMT.

         - Access the IVE_FFP table and Select IVE_FFP.am_ffp WHERE IVE_FFP.ts_cr is most recent and IVE_FFP.dt_eff <= PAYMENT.dt_pmnt_begin (this will pick up the most recently entered IV-E FFP percentage that was effective at the

time of service begin date).

- Multiply IVE_FFP.am_ffp with WS-FED-AMT.

- Update PAYMENT_REIMBURSEMENT.am_reimb with the result

4.1.6. PAYMENT_REIMBURSEMENT.am_titlexix = NULL.

4.1.7. PAYMENT_REIMBURSEMENT.am_ffp_pct= Refer to PAYMENT_REIMBURSEMENT.am_reimb field for the details of the retrieval.

4.1.8. PAYMENT_REIMBURSEMENT.am_titlexix_pct = NULL.

4.1.9. PAYMENT_REIMBURSEMENT.am_maint_pct = NULL.

4.2. The **second PAYMENT_REIMBURSEMENT record** is populated as described below.

4.2.1. PAYMENT_REIMBURSEMENT.id_pmnt = PAYMENT.id_pmnt

4.2.2. PAYMENT_REIMBURSEMENT.ts_cr = Current system date

4.2.3. PAYMENT_REIMBURSEMENT.id_cr = Batch ID.

4.2.4. PAYMENT_REIMBURSEMENT.cd_reimb_type = "2" (i.e., State)

4.2.5. PAYMENT_REIMBURSEMENT.am_reimb = (PAYMENT.am_nontrust – PAYMENT_REIMBURSEMENT.am_reimb of the first record inserted). Set this field to [(PAYMENT.am_rqst – am_admin) – PAYMENT_REIMBURSEMENT of the first record inserted] for (-) payments.

4.2.6. PAYMENT_REIMBURSEMENT.am_titlexix = NULL.

4.2.7. PAYMENT_REIMBURSEMENT.am_ffp_pct= NULL.

4.2.8. PAYMENT_REIMBURSEMENT.am_titlexix_pct = NULL.

4.2.9. PAYMENT_REIMBURSEMENT.am_maint_pct = NULL.

4.3. Update the selected PAYMENT record with PAYMENT.fl_ive_calc = "N."

4.4. Continue from (4) until all payments are processed.

4.5. The PAYMENT_REIMBURSEMENT record is populated as described below.

4.5.1. PAYMENT_REIMBURSEMENT.id_pmnt = PAYMENT.id_pmnt

4.5.2. PAYMENT_REIMBURSEMENT.ts_cr = Current system date

4.5.3. PAYMENT_REIMBURSEMENT.id_cr = Batch ID.

4.5.4. PAYMENT_REIMBURSEMENT.cd_reimb_type = "2" (i.e., State)

4.5.5. PAYMENT_REIMBURSEMENT.am_reimb = PAYMENT.am_nontrust ((PAYMENT.am_rqst – PAYMENT.am_admin for (-) payments)

4.5.6. PAYMENT_REIMBURSEMENT.am_titlexix = NULL.

4.5.7. PAYMENT_REIMBURSEMENT.am_ffp_pct= NULL.

4.5.8.   PAYMENT_REIMBURSEMENT.am_titlexix_pct = NULL.

4.5.9.   PAYMENT_REIMBURSEMENT.am_maint_pct = NULL.

4.6. Update the selected PAYMENT record with PAYMENT.fl_ive_calc = "N."

4.7. Whenever a PAYMENT_REIMBURSEMENT record is inserted, set PAYMENT_REIMBURSEMENT.cd_stat = "A." Set the cd_stat = "I" on all other PAYMENT_REIMBURSEMENT records where PAYMENT_REIMBURSEMENT.id_pmnt = PAYMENT.id_pmnt.

4.8. Continue from (4).


**Section D (Retroactive run)**

This section of the program re-calculates the Title IV-E eligible percentages if (1) the FFP percentage changes retroactively and/or (2) the provider's eligible cost percentages change retroactively. To do this, the batch program reads ONLINE_TRIGGER table and selects those records where ONLINE_TRIGGER.cd_txn = 50 and those record WHERE ONLINE_TRIGGER.cd_txn = 51.

The batch program processes all the selected triggers as described below.

**Processing for ONLINE_TRIGGER.cd_txn = 50**

Select all PAYMENT records where PAYMENT.dt_bgn >= ONLINE_TRIGGER.ts_efct_frm. UPDATE PAYMENT.fl_ive_calc = "Y" AND continue from Section A of this batch program. Delete the ONLINE_TRIGGER record processed.

**Processing for ONLINE_TRIGGER.cd_txn = 51**

Select all PAYMENT records where PAYMENT.dt_bgn >= ONLINE_TRIGGER.ts_efct_frm AND PAYMENT.id_payee = ONLINE_TRIGGER.id_trigger1 AND PAYMENT.cd_srvc = ONLINE_TRIGGER.id_trigger2. UPDATE PAYMENT.fl_ive_calc = "Y" AND continue from Section A of this batch program. . Delete the ONLINE_TRIGGER record processed.

### 1.4.4. <u>Calculate Eligibility-Based Days of Care Batch Program</u>

**Program Name:**    b-fm04a11-calc-days-of-care

**Process Summary:**    The 'Calculate Eligibility-Based Days of Care' batch program divides the children's placements and gaps in placement, which fall within an eligibility episode, into monthly periods. For every period, the program inserts a row into the new 'Days_of_Care_Period' table. It uses logic similar to that used by the 'Identify FFP/Non-FFP Payments' batch program to calculate the number of days earmarked for a particular eligibility status and stores them in Qt_elig_reimb, Qt_inelig, Qt_elig_nonreimb, Qt_elig_ssi and Qt_pending. Id_epsd links the placement period to the placement record in WiSACWIS. Id_elig links the placement and gap periods to the child's eligibility record. Dt_begin and dt_end are always in the same month. Cd_stat has two possible values, 'A' for Active and 'I' for Inactive. For placement periods, cd_cnty is derived from the county associated with the placement (Episode.cd_cnty). For gap periods, cd_cnty is derived from the county of the child's last placement in that case. Fl_last is used by the batch program to designate the last processed period in an open placement or gap in placement.

Every month, the program re-calculates the days of care for eligibility records that were affected by certain online events such as the completion of an eligibility determination and backdating the ending of a placement (Section A). In addition, the program adds a new placement or gap period for WiSACWIS current open eligibility records. (Section B).

On a monthly basis, the 'Eligibility-Based Days of Care' report will provide the users with counts of days of care. For details, refer to the 'Eligibility-Based Days of Care' report design document.

Finally, the 'Eligibility-Based Days of Care' page will be available to view the 'Days_of_Care_Period' records (both active and inactive) and to download those records into a spreadsheet for further analysis. For details, refer to the 'Eligibility-Based Days of Care' developer's guide.

**Frequency:**    Monthly with MF3..

**Dependencies:**    None.

**Input Parameters:**    Batch Program Number, Month Processed

**Input Files:**        fm04a11-b-parm

**Output Files:**       Fm04a11-b-{date time}.LOG

**Database Tables:**    Days_of_Care_Period      CRU

                                     Eligibility              R

                                     Elig_Redet               R

                                     Episode                  R

                                     Online_Trigger           RU

### Process Description:

### Section A:

- Select all the rows in the Online_Trigger table with Online_Trigger.cd_txn= 7000 and Online_Trigger.dt_batch_prcs= Null,

    - If there is a row with Online_Trigger.id_trigger= Null, write to the error log (Missing eligibility id in trigger record- Trigger ID= 'id_online_trigger') and skip this Online_Trigger row.

    - If Online_Trigger.id_trigger1 is Null, write to the error log (Missing case id in trigger record- Trigger ID= 'id_online_trigger') and skip this Online_Trigger row.

    - If Online_Trigger.id_trigger2 is Null, write to the error log (Missing person id in trigger record- Trigger ID= 'id_online_trigger') and skip this Online_Trigger row.

    - If there is a row with Online_Trigger. ts_efct_frm = Null, write to the error log (Missing effective date in trigger record- Trigger ID= 'id_online_trigger') and skip this Online_Trigger row.

    - If there is more than one row with the same value in Online_Trigger.id_trigger-  for each of the rows- write to the error log (Multiple trigger records with the same eligibility id- Trigger ID= 'Online_Trigger.id_online_trigger', Eligibility ID= 'Online_Trigger.id_trigger) and skip these Online_Trigger rows.


    -       Select Days_of_Care_Period row where

Days_of_Care_Period.id_elig= Online_Trigger.id_trigger and
Days_of_Care_Period.id_case= Online_Trigger.id_trigger1 and
Days_of_Care_Period.id_prsn= Online_Trigger.id_trigger2 and
Days_of_Care_Period.cd_stat = 'A' and

Days_of_Care_Period.fl_last='Y' and

Days_of_Care_Period.dt_end is prior to the month entered on the parameter file.

If there is more than one row which meet the criteria above and have the same value in Days_of_Care_Period.id_elig, write to the error log (Multiple last period records with the same eligibility id– PERIOD ID= 'Days_of_Care_Period.id_period', Eligibility ID= 'Days_of_Care_Period.id_elig') and skip this Online_Trigger row.

### **Section B:**

- Select all the rows in the Days_of_care_Period table with Days_of_Care_Period.fl_last='Y' and Days_of_Care_Period.cd_stat = 'A' and dt_end is prior to the month entered on the parameter file:

  - If there is more than one row which meet the criteria above and have the same value in Days_of_Care_Period.id_elig, write to the error log (Multiple last period records with the same eligibility id– PERIOD ID= 'Days_of_Care_Period.id_period', Eligibility ID= 'Days_of_Care_Period.id_elig') and skip these Days_of_Care_Period rows.

Step 1: Determine the eligibility id:

- When processing Online_Trigger rows:

  - Set ELIGIBILITY ID to Online_Trigger.id_trigger

  - If Eligibility.fl_void= 'Y' or Eligibility.fl_deter_na= 'Y' where Eligibility.id_elig= ELIGIBILITY ID, go to Step 7.

- When processing current placements and gaps:

  - Set ELIGIBILITY ID to Days_of_Care_Period.id_elig

  - If Eligibility.fl_void= 'N' or Eligibility.fl_deter_na= 'Y' where Eligibility.id_elig= ELIGIBILITY ID, then go to Step 7.

Step 2: Determine the case id:

- When processing Online_Trigger rows:

  - Set CASE ID to Online_Trigger.id_trigger1

- When processing current placements and gaps:

  - Set CASE ID to Days_of_Care_Period.id_case

Step 3: Determine the person id:

- When processing Online_Trigger rows:

- Set PERSON ID to Online_Trigger.id_trigger2


- When processing current placements and gaps:

    • Set PERSON ID to Days_of_Care_Period.id_prsn


Step 4: Determine the start date:

- When processing Online_Trigger rows:

    • Select Days_of_Care_Period row where

    Days_of_Care_Period.id_elig= Online_Trigger.id_trigger and
    Days_of_Care_Period.id_case= Online_Trigger.id_trigger1 and
    Days_of_Care_Period.id_prsn= Online_Trigger.id_trigger2 and
    Days_of_Care_Period.cd_stat = 'A' and

    Max(Days_of_Care_Period.dt_end)


    • If there is only one row selected, which meets this criteria,

        - Set START DATE to one day after Days_of_Care_Period.dt_end

        - Set Days_of_Care.fl_last= 'N', Select Days_of_Care_Period row where

            Days_of_Care_Period.id_elig= Online_Trigger.id_trigger and
            Days_of_Care_Period.id_case= Online_Trigger.id_trigger1 and
            Days_of_Care_Period.id_prsn= Online_Trigger.id_trigger2 and
            Days_of_Care_Period.cd_stat = 'A' and

            Max(Days_of_Care_Period.dt_end)


    • If no row is found in the Days of Care Period that meets the criteria above:

        - Set START DATE to Online_Trigger.ts_efct_frm, if

            (Eligibility.fl_cmplt= 'N' and

            Eligibility.dt_bgn <= Online_Trigger.ts_efct_frm) OR

            (Eligibility.fl_cmplt= 'Y' and Eligibility.dt_eff <= Online_Trigger.ts_efct_frm),
            where Eligibility.id_elig-Online_Trigger.id_trigger.


        - Otherwise, set START DATE to Eligibility.dt_bgn if Eligibility.fl_cmplt= 'N'. Set
          START DATE to Eligibility.dt_eff if Eligibility.fl_cmplt= 'Y'.


- When processing current placements and gaps:

- Set START DATE to one day after Days_of_Care_Period.dt_end.

    - Set Days_of_Care.fl_last= 'N', Select Days_of_Care_Period row where

        Days_of_Care_Period.id_elig= Online_Trigger.id_trigger and
        Days_of_Care_Period.id_case= Online_Trigger.id_trigger1 and
        Days_of_Care_Period.id_prsn= Online_Trigger.id_trigger2 and
        Days_of_Care_Period.cd_stat = 'A' and

        Max(Days_of_Care_Period.dt_end)


Step 5: Determine the end date:


- CLOSED ELIGIBILITY = Y

- If Eligibility.dt_end is not null where Eligibility.id_elig = ELIGIBILITY ID


    - If Eligibility.dt_end <= last day of the month entered on the parameter file, set  END DATE to Eligibility.dt_end


    - If  Eligibility.dt_end > last day of the month entered on the parameter file, set  END DATE to the last day of the month entered on the parameter file and set CLOSED ELIGIBILITY = N


- Else, if Elig_Redet.dt_end is not null where Elig_Redet.id_elig = ELIGIBILITY ID

    - If Elig_Redet.dt_end <= last day of the month entered on the parameter file, set  END DATE to Elig_Redet.dt_end


    - If  Elig_Redet.dt_end > last day of the month entered on the parameter file, set  END DATE to the last day of the month entered on the parameter file and set CLOSED ELIGIBILITY = N


- Else, if Elig_Redet.dt_end is null where Elig_Redet.id_elig = ELIGIBILITY ID

    - Set END DATE to the last day of the month entered on the parameter file and set CLOSED ELIGIBILITY = N.


Step 5.5: Validate Dates

- If START DATE > END DATE- eligibility has ended or bad data:

    - If CLOSED ELIGIBILITY, go to Step 7.

- If not CLOSED ELIGIBILITY, write to the error log (Invalid eligibility period start and end dates- Eligibility ID = 'ELIGIBILITY ID') and skip the next steps.

Step 6: Process Days of Care Periods:

- Set MONTH PROCESSED to Month( START DATE).

- Set FIRST PERIOD = 'Y'

- Set LAST PERIOD = 'N'

- While not LAST PERIOD do steps 6.1 through 6.4 below:

Step 6.1: Determine the period start and end date:

- Set PERIOD START DATE to START DATE, if FIRST PERIOD.

- Otherwise, Set PERIOD START DATE to one day after the PERIOD END DATE of the last period processed.

- Set FIRST PERIOD to 'N'.

- Use ID CASE and ID PERSON to retrieve the child's placement in which the PERIOD START DATE falls. Select Episode where Episode.id_case= CASE ID and Episode.id_prsn = PERSON ID and (Episode.cd_epsd_type= 1 (OOH) or (Episode.cd_epsd_type= 2 (IH)and Service_Type.cd_srvc_ctgry= 503 (AAFC Adoption) where Episode.cd_srvc = Service_Type.cd_srvc) and Episode.end_reason <> code value for 'Placement Made in Error' and Episode.dt_begin < = PERIOD START DATE and (Episode.dt_end is null or Episode.dt_end >= PERIOD START DATE) and there is a final approval row for this placement.

I. If no valid placement is retrieved (i.e. the period starts with a gap in placement)

   - Set EPISODE ID to null

   1. If last day in MONTH PROCESSED >= END DATE

      A. If there are no placement records retrieved where Episode.id_case= CASE ID and Episode.id_prsn = PERSON ID and (Episode.cd_epsd_type= 1 (OOH) or (Episode.cd_epsd_type= 2 (IH)and Service_Type.cd_srvc_ctgry= 503 (AAFC Adoption) where Episode.cd_srvc = Service_Type.cd_srvc) and Episode.end_reason <> code value for 'Placement Made in Error' and Episode.dt_begin > PERIOD START DATE and Episode.dt_begin < = END DATE )- this gap in placement is the last period to process

         - Set PERIOD END DATE to END DATE

- Set LAST PERIOD to 'Y'

B. If there is at least one placement record retrieved where Episode.id_case= CASE ID and Episode.id_prsn = PERSON ID and (Episode.cd_epsd_type= 1 (OOH) or (Episode.cd_epsd_type= 2 (IH)and Service_Type.cd_srvc_ctgry= 503 (AAFC Adoption) where Episode.cd_srvc = Service_Type.cd_srvc) and Episode.end_reason <> code value for 'Placement Made in Error' and Episode.dt_begin > PERIOD START DATE and Episode.dt_begin < = END DATE) and there is a final approval row for this placement- this gap in placement is not the last period to process

  - Set PERIOD END DATE to one day prior to the earliest Episode.dt_begin of those placements.

2. If last day in MONTH PROCESSED < END DATE

   A. If there are no placement records retrieved where Episode.id_case= CASE ID and Episode.id_prsn = PERSON ID and (Episode.cd_epsd_type= 1 (OOH) or (Episode.cd_epsd_type= 2 (IH)and Service_Type.cd_srvc_ctgry= 503 (AAFC Adoption) where Episode.cd_srvc = Service_Type.cd_srvc) and Episode.end_reason <> code value for 'Placement Made in Error' and Episode.dt_begin > PERIOD START DATE and Episode.dt_begin < = last day in MONTH PROCESSED) and there is a final approval row for this placement- this gap in placement is the last period in the processed month

     - Set PERIOD END DATE to last day in MONTH PROCESSED.

     - Increase MONTH PROCESSED by one month

   B. If there is at least one placement record retrieved where Episode.id_case= CASE ID and Episode.id_prsn = PERSON ID and (Episode.cd_epsd_type= 1 (OOH) or (Episode.cd_epsd_type= 2 (IH)and Service_Type.cd_srvc_ctgry= 503 (AAFC Adoption) where Episode.cd_srvc = Service_Type.cd_srvc) and Episode.end_reason <> code value for 'Placement Made in Error' and Episode.dt_begin > PERIOD START DATE and Episode.dt_begin < = last day in MONTH PROCESSED) and there is a final approval row for this placement- this gap in placement is NOT the last period in the month processed

     - Set PERIOD END DATE to one day prior to the earliest Episode.dt_begin of those placements.

II. If one placement is retrieved (i.e. the period starts with a placement)

   - Set EPISODE ID to Episode.id_epsd

   1. If last day in MONTH PROCESSED >= END DATE

A. If Episode.dt_end >= END DATE  or Episode.dt_end is Null

- Set PERIOD END DATE to END DATE

- Set LAST PERIOD to 'Y'


B. If Episode.dt_end < END DATE

- Set PERIOD END DATE to Episode.dt_end


2. If last day in MONTH PROCESSED < END DATE

A. If Episode.dt_end >= last day in MONTH PROCESSED

or Episode.dt_end is Null

- Set PERIOD END DATE to last day in MONTH PROCESSED.

- Increase MONTH PROCESSED by one month


B. If Episode.dt_end < last day in MONTH PROCESSED

- Set PERIOD END DATE to Episode.dt_end


III. If more than one placement is retrieved (i.e. overlapping placements), write to the error log (Overlapping placements- Eligibility ID = 'ELIGIBILITY ID') and skip the next steps.


Step 6.2: Determine the number of days within each eligibility status:

- If Eligibility.fl_cmplt = 'N' where Eligibility.id_elig = ELIGIBILITY ID, then:

  - Set ER DAYS to 0

  - Set ENR DAYS to 0

  - Set IE DAYS to 0

  - Set SSI DAYS to 0

  - Set PENDING DAYS to 'PERIOD END DATE minus PERIOD START DATE +1'

  - Go to Step 6.3.


- Else, if Eligibility.fl_cmplt = 'Y' where Eligibility.id_elig = ELIGIBILITY ID, then:

  - Go to Step 6.2.1.

Step 6.2.1:

- Select the Eligibility records where Eligibility.dt_eff (or Elig_Redet.dt_eff)<= PERIOD START DATE and Eligibility.dt_cmplt (or Elig_Redet.dt_cmplt) is most recent and Eligibility.id_elig = ELIGIBILITY ID. Store dt_eff, cd_stat and cd_ssi_status in a temporary table.

- Continue to execute the following routine while Eligibility.dt_eff (or Elig_Redet.dt_eff) of the processed Eligibility record <= PERIOD END DATE. Once all Eligibility records are processed for this period, continue with Step 6.2.3.

  - Select the next Eligibility record and store Eligibility.dt_eff (or Elig_Redet.dt_eff), Eligibility.cd_stat (Elig_Rdet.cd_stat) and Eligibility.cd_ssi_status (or Elig_Redet.cd_ssi_status) in the same temporary table ONLY if satisfies all the criteria listed below. If it fails at least one criterion, then do not store the Eligibility record information and continue to process the next Eligibility record.

    - Eligibility.dt_eff (or Elig_Redet.dt_eff) of the most recently stored Eligibility record < ELIGIBILITY.dt_eff (or Elig_Redet.dt_eff) of the currently processed Eligibility record <= PERIOD END DATE **AND**

    - Eligibility.dt_cmplt (or Elig_Redet.dt_cmplt) of the currently processed Eligibility record > Eligibility.dt_cmplt (or ELIG_REDET.dt_cmplt) of the most recently stored Eligibility record.

- If there is NO Eligibility row where eligibility.dt_eff (or Elig_Redet.dt_eff) <= PERIOD START DATE, but there is at least one Eligibility row where Eligibility.dt_eff (or Elig_Redet.dt_eff) <= PERIOD END DATE, then continue with 6.2.2.

Step 6.2.2

- Since there are NO eligibility records that have Eligibility.dt_eff (or Elig_Redet.dt_eff) <= PERIOD START DATE, the portion of the period that falls from PERIOD START DATE to Eligibility.dt_eff (or Elig_Redet.dt_eff) of the first available Eligibility record will be assumed as being "Pending." Therefore, store PERIOD START DATE and cd_stat = "4" (i.e., Pending) at a temporary table.

- Select the Eligibility record where Eligibility.dt_eff (or Elig_Redet.dt_eff) is least recent and Eligibility.dt_eff (or Elig_Redet.dt_eff)<= PERIOD END DATE. If there are more than one ELIGIBILITY or ELIG_REDET records with the same dt_eff, then select the record with the most recent Eligibility.dt_cmplt (or Elig_Redet.dt_cmplt).

- Continue to execute the following routine while Eligibility.dt_eff (or Elig_Redet.dt_eff) (of the processed Eligibility record) <= PERIOD END DATE. Once all Eligibility rows are processed for this period, continue with Step 6.2.3.

  - Select the next Eligibility record and store Eligibility.dt_eff (or Elig_Redet.dt_eff) and ELIGIBILITY.cd_stat (or ELIG_REDET.cd_stat) and Eligibility.cd_ssi_status (or Elig_Redet.cd_ssi_status) in the same temporary table ONLY if is satisfies all the criteria listed below. If it fails at least one criterion, then do not store the Eligibility

record information and continue to process the next Eligibility record.

- Eligibility.dt_eff (or Elig_Redet.dt_eff) of the most recently stored Eligibility record < Eligibility.dt_eff (or Elig_Redet.dt_eff) of the currently processed Eligibility record <= PERIOD END DATE AND

- Eligibility.dt_cmplt (or Elig_Redet.dt_cmplt)of the currently processed Eligibility record > Eligibility.dt_cmplt (or Elig_Redet.dt_cmplt)of the most recently stored Eligibility record.

Step 6.2.3

- Once steps 6.2.1 and 6.2.2 are executed, the result set should be stored in a temporary table with three columns. The first column stores the effective dates, the second column stores the 'eligibility and reimbursability' status of the child as of that date, and the third column stores the SSI status (for administrative claiming purposes) as of that date.

- Based on the way steps 6.2.1 and 6.2.2 are executed, the effective dates should be sorted by a chronological order. The following example shows a possible result set for a period with PERIOD START DATE = 03/01/2000 and PERIOD END DATE = 03/31/2000.

| Effective Date | Eligibility Status | SSI Status |
|---|---|---|
| 03/01/2000 | 1 (i.e., Eligible and Reimbursable) | N |
| 03/15/2000 | 3 (i.e., Eligible but Non-reimbursable) | Y |
| 03/20/2000 | 1 (i.e., Eligible and Reimbursable) | N |
| 03/26/2000 | 3 (i.e., Eligible but Non-reimbursable) | N |

- The child with the above result set will have the following statuses on the Days_of_Care_Period table:

| Effective Dates | Eligibility Status |
|---|---|
| PERIOD START DATE – 03/15/2000 | 1 (i.e., Eligible and Reimbursable) |
| 03/15/2000 – 03/20/2000 | 5 (i.e., Eligible and Receiving SSI) - Note that the status has changed from 3 to 5 because of SSI. |

| 03/20/2000 – 3/26/2000 | 1 (i.e., Eligible and Reimbursable) |
|---|---|
| 03/26/2000 – PERIOD END DATE and add one day | 3 (i.e., Eligible but Non-reimbursable. Not receiving SSI) |

- If only one row is selected from the Eligibility (or Elig_Redet) table then the calculation will be PERIOD END DATE – PERIOD START DATE + 1

- When calculating the "Eligible and Reimbursable", "Eligible but Non-reimbursable", "Ineligible", 'Eligible and Receiving SSI' and "Pending" portions of a period, the number of days that the child spent with these statuses are calculated. If we use the above example, then the following table depicts the number of days for the child in each status.

| Status | Number of Days |
|---|---|
| Eligible and Reimbursable | 20 |
| Eligible but Non-reimbursable (Not Receiving SSI). | 5 |
| Eligible but Receiving SSI | 6 |

- Set ER DAYS to the number of days with cd_stat = "1."

- Set ENR DAYS to the number of days with cd_stat = "2."

- Set IE DAYS to the number of days with cd_stat = "3."

- Set SSI DAYS to the number of days with cd_stat = "5."

- Set PENDING DAYS to the number of days with cd_stat = "4."

Step 6.3: Determine the county code:

- If EPISODE ID is not null (i.e. placement period)

  - Set COUNTY CODE to Episode.cd_cnty where Episode.id_epsd= EPISODE ID

- If EPISODE ID is null (i.e. gap in placement period)

  - Set COUNTY CODE to Episode.cd_cnty where Episode.id_prsn = PERSON ID and Episode.id_case= CASE ID and Episode.dt_end < PERIOD START DATE and max(Episode.dt_end) and (Episode.cd_epsd_type= 1 (OOH) or (Episode.cd_epsd_type= 2 (IH)and Service_Type.cd_srvc_ctgry= 503 (AAFC Adoption) where Episode.cd_srvc = Service_Type.cd_srvc) and Episode.end_reason <> code value for 'Placement Made in

Error' and there is an final approval row for this placement- child's last placement prior to the start of the gap in placement.

- If more than one placement is retrieved (i.e. overlapping placements), write to the error log (Overlapping placements- Eligibility ID = 'ELIGIBILITY ID') and skip the next steps.

- If there is no placement found that meets this criteria, set COUNTY CODE to Episode.cd_cnty where Episode.id_prsn = PERSON ID and Episode.id_case= CASE ID and Episode.dt_begin > PERIOD END DATE and min(Episode.dt_begin) and (Episode.cd_epsd_type= 1 (OOH) or (Episode.cd_epsd_type= 2 (IH)and Service_Type.cd_srvc_ctgry= 503 (AAFC Adoption) where Episode.cd_srvc = Service_Type.cd_srvc) and Episode.end_reason <> code value for 'Placement Made in Error' and there is an final approval row for this placement- child's earliest placement after the end of the gap in placement..

- If more than one placement is retrieved (i.e. overlapping placements), write to the error log (Overlapping placements- Eligibility ID = 'ELIGIBILITY ID') and skip the next steps.

- If there is no placement found that meets this criteria, write to the error log (No valid placement exists- Eligibility ID = 'ELIGIBILITY ID') and skip the next steps.

Step 6.4: Insert a row into the Days_of_Care_Period:

Days_of_Care_Period.Id_period=  Unique table ID

Days_of_Care_Period.Id_elig=  ELIGIBILITY ID

Days_of_Care_Period.Id_epsd=  EPISODE ID

Days_of_Care_Period.Id_case= CASE ID

Days_of_Care_Period.Id_prsn= PERSON ID

Days_of_Care_Period.Dt_begin= PERIOD START

Days_of_Care_Period.Dt_end= PERIOD END

Days_of_Care_Period.Qt_elig_reimb= ER DAYS

Days_of_Care_Period.Qt_inelig=IE DAYS

Days_of_Care_Period.Qt_elig_nonreimb= ENR DAYS

Days_of_Care_Period.Qt_elig_ssi= SSI DAYS

Days_of_Care_Period.Qt_pending= PENDING DAYS

Days_of_Care_Period.Cd_stat= 'A'

Days_of_Care_Period.Fl_last= LAST PERIOD

Days_of_Care_Period.Cd_cnty= COUNTY CODE

Days_of_Care_Period.Ts_cr= current system date and time

Days_of_Care_Period.Id_cr= Batch ID

Days_of_Care_Period.Ts_up= current system date and time

Days_of_Care_Period.Id_up= Batch ID

Step 7:

-   When processing Online_Trigger rows:

    •   Set Online_Trigger.dt_batch_prcs = current system date and time.

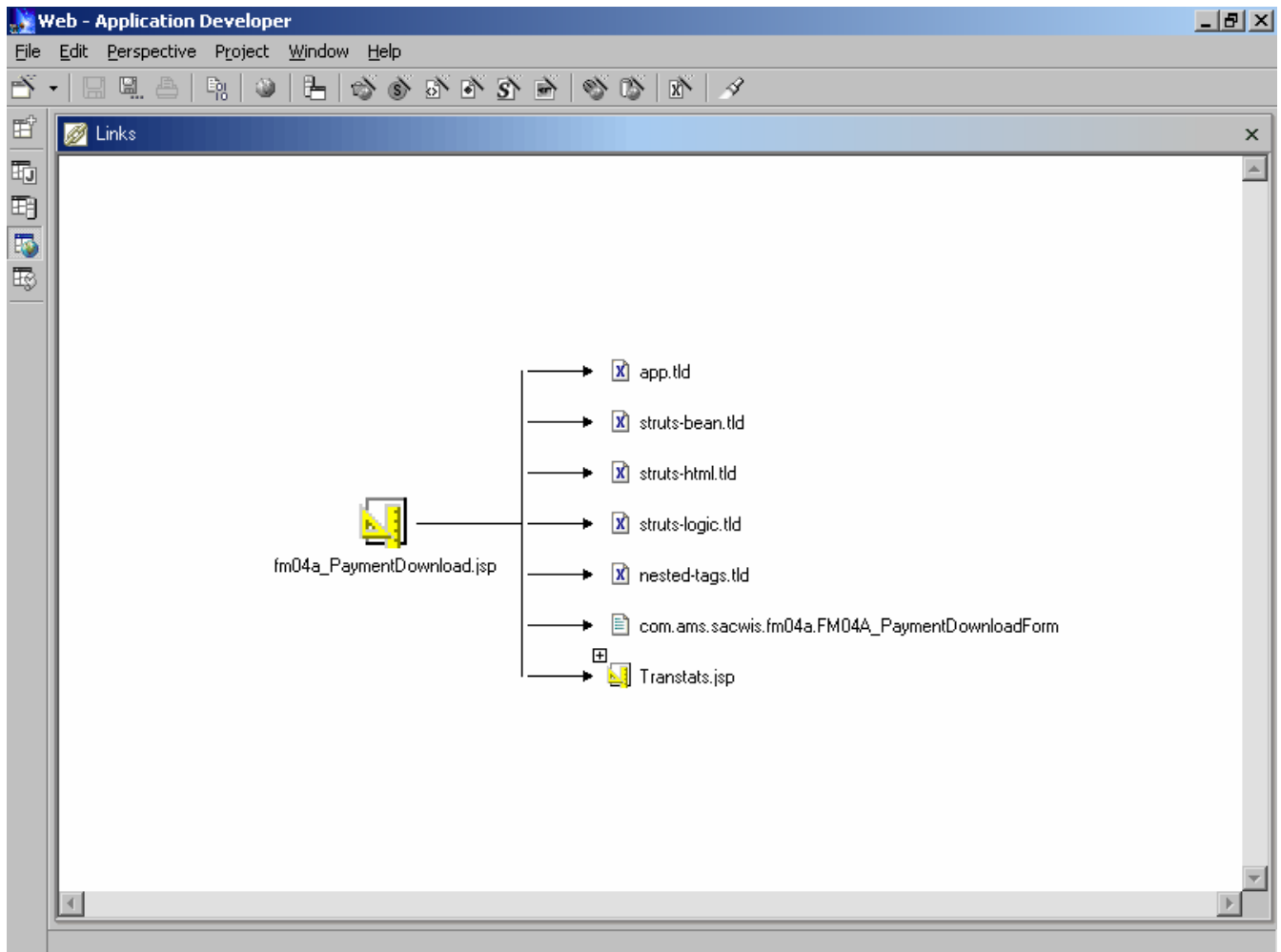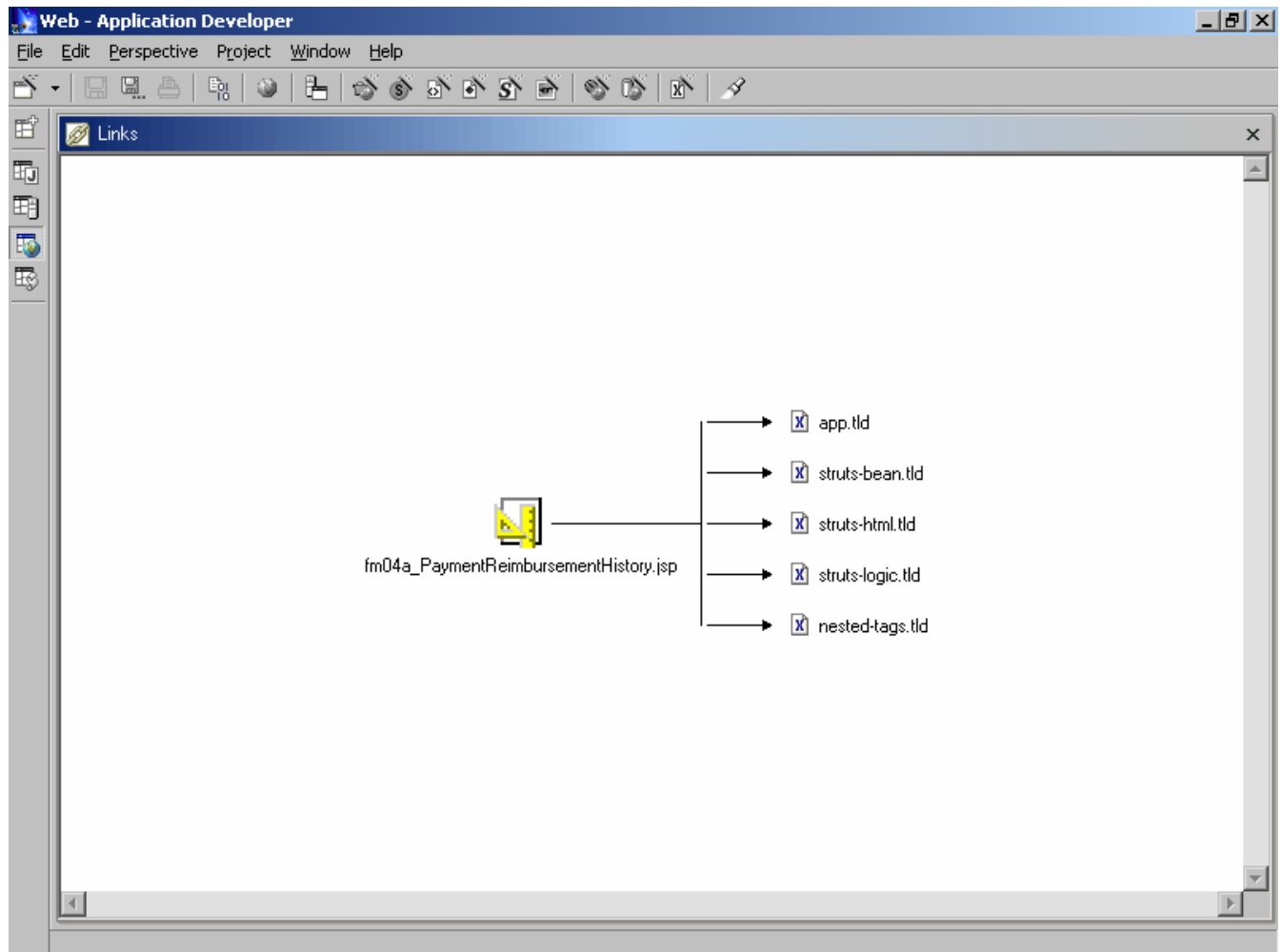## 1.5.    Online Components

### 1.5.1.    User Interface Components

There are two JSPs for the FM04a topic, fm04a_PaymentDownload.jsp and fm04a_PaymentReimbursementHistory.jsp.  Each has a corresponding JavaScript file.

Below are the Link diagrams of these pages.

**fm04a_PaymentDownload.jsp**

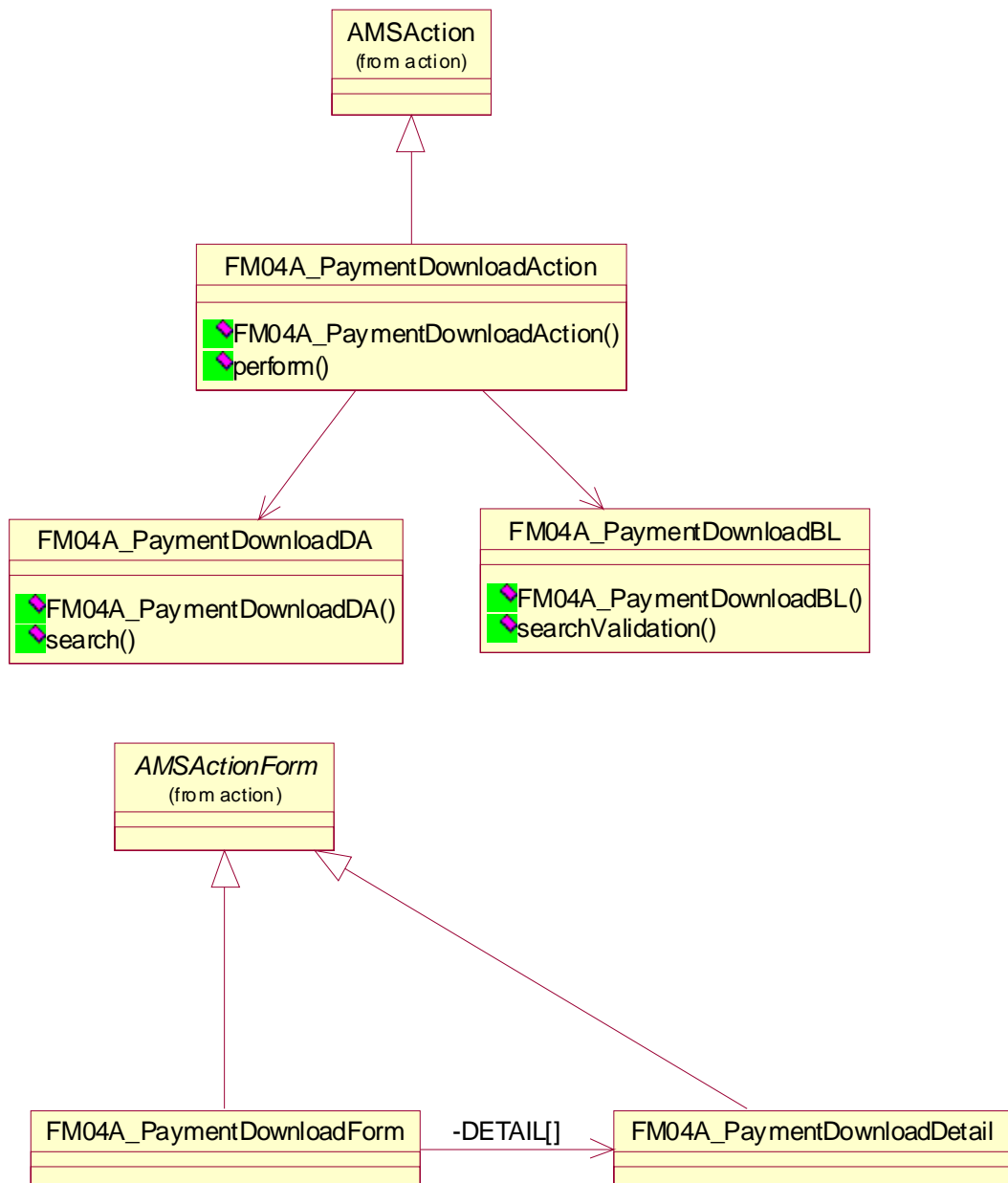## fm04a_PaymentReimbursementHistory.jsp

*eWiSACWIS Detailed Design Document*

### 1.5.2.  <u>Java Components</u>

Below are the Class diagrams for the FM04a topic.  There are two distinct "families".  One that services all the Payment download transactions and another that services all the Payment Reimbursement History transactions.

Main Classes involved in Payment Download transactions.



*FM04a- 77*

Main Classes involved in Payment Reimbursement History transactions.
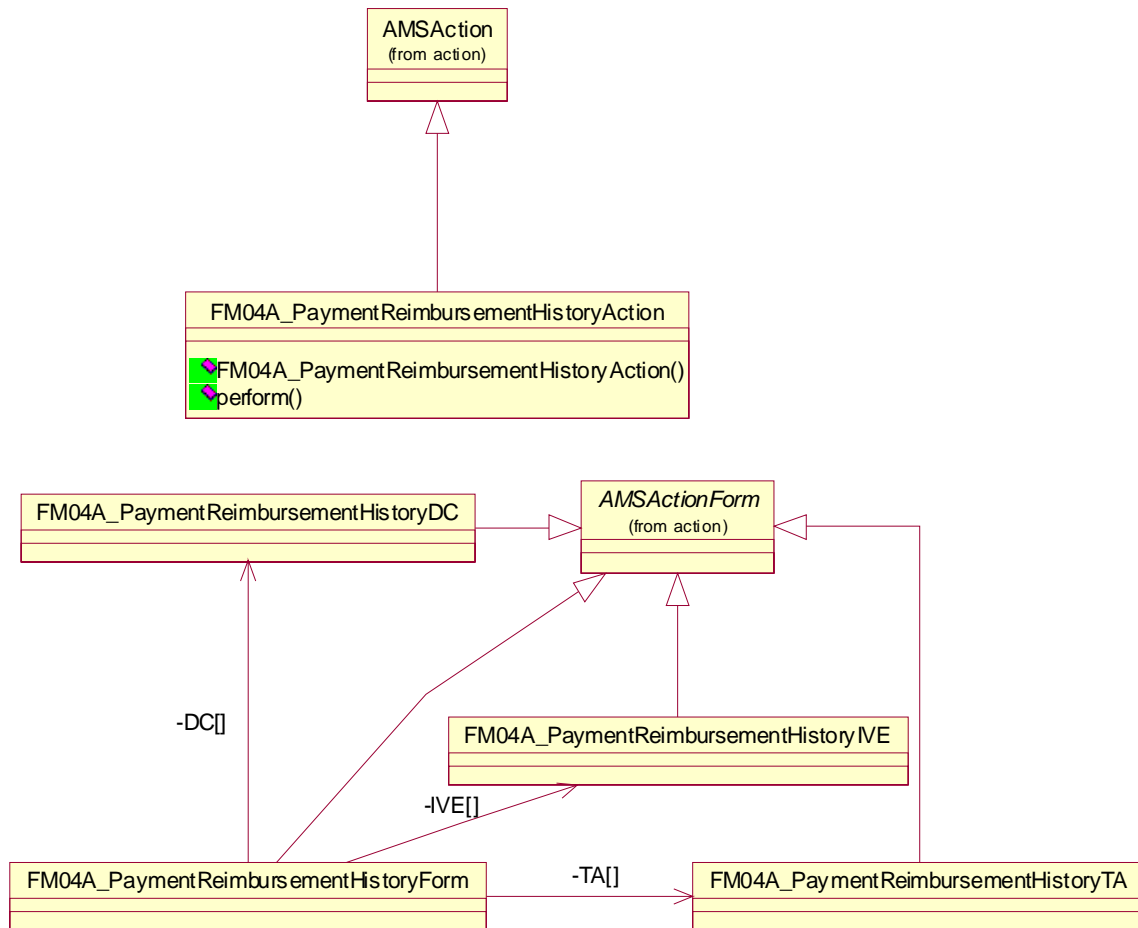
Below is a brief description of each of the Java Classes:

| Type | Name | Comment |
| --- | --- | --- |
| Action Class | FM04A_PaymentDownloadAction | Standard Action Class. |
| Data Access | FM04A_PaymentDownloadDA | Standard Data Access Class. |
| Business Logic | FM04A_PaymentDownloadBL | Standard Business Logic Class contains logic to validate the payment search. |
| Form | FM04A_PaymentDownloadForm | Aggregate Data container for Payment Download information. |
| Form | FM04A_PaymentDownloadFormDetail | Sub form of Payment Download Form. Data container for Payment Download Detail information |
| Action Class | FM04A_PaymentReimbursementHistoryAction | Standard Action Class. |
| Data Access | FM04A_PaymentReimbursementHistoryDA | Standard Data Access Class. |
| Form | FM04A_PaymentReimbursementHistoryForm | Aggregate Data container for Payment Reimbursement History information. |
| Form | FM04A_PaymentReimbursementHistoryDC | Sub form of Payment Reimbursement History Form. Data container for Payment Reimbursement History DC information |
| Form | FM04A_PaymentReimbursementHistoryIVE | Sub form of Payment Reimbursement History Form. Data container for Payment Reimbursement History IVE information |
| Form | FM04A_PaymentReimbursementHistoryTA | Sub form of Payment Reimbursement History Form. Data container for Payment Reimbursement History TA information |